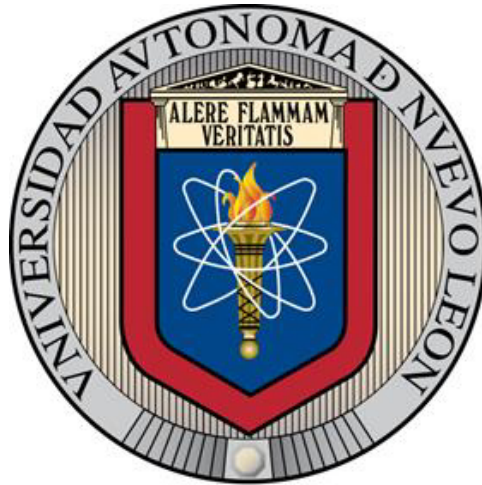


UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA



**DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO
PARA EL VUELO EN FORMACIÓN DE VEHÍCULOS
AÉREOS NO TRIPULADOS**

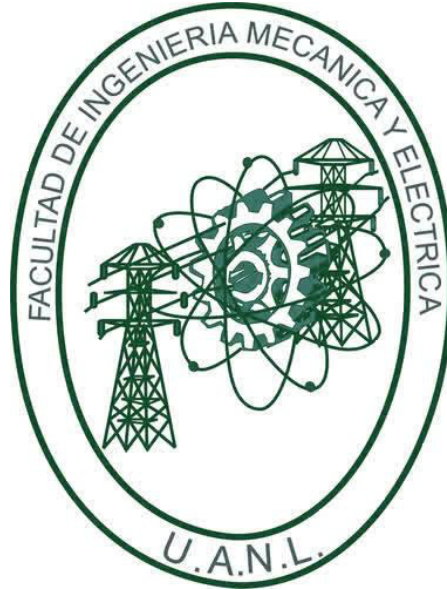
POR

ERIK GILBERTO ROJO RODRÍGUEZ

**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
MAESTRÍA EN INGENIERÍA AERONÁUTICA
CON ORIENTACIÓN EN DINÁMICA DE VUELO**

AGOSTO, 2017

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO**



**DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO
PARA EL VUELO EN FORMACIÓN DE VEHÍCULOS
AÉREOS NO TRIPULADOS**

POR

ERIK GILBERTO ROJO RODRÍGUEZ

**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
MAESTRÍA EN INGENIERÍA AERONÁUTICA
CON ORIENTACIÓN EN DINÁMICA DE VUELO**

AGOSTO, 2017

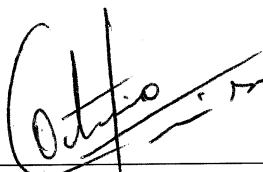
Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Subdirección de Estudios de Posgrado

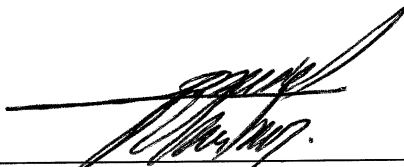
Los miembros del Comité de Tesis recomendamos que la Tesis «Diseño e implementación de un algoritmo para el vuelo en formación de vehículos aéreos no tripulados», realizada por el alumno Erik Gilberto Rojo Rodríguez, con número de matrícula 1492357, sea aceptada para su defensa como requisito parcial para obtener el grado de Maestría en Ingeniería Aeronáutica con orientación en Dinámica de Vuelo.

El Comité de Tesis



Dr. Octavio García Salazar

Asesor



M.C. Daniel Librado Martínez Vázquez

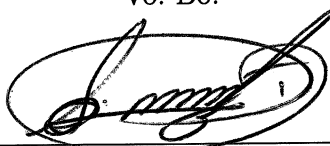
Revisor



Dr. Anand Eleazar Sánchez Orta

Revisor

Vo. Bo.



Dr. Simón Martínez Martínez

Subdirección de Estudios de Posgrado

San Nicolás de los Garza, Nuevo León, agosto 2017

*A mis padres, María de la Luz y Gilberto,
que siempre han sabido como guiarme por el camino correcto,
y porque gracias a ellos he cumplido mis metas.*

Índice general

Índice de figuras	IX
Índice de tablas	XV
Nomenclatura	XVI
Agradecimientos	XXI
Resumen	XXIII
1. Introducción	1
1.1. Vehículos Aéreos No Tripulados	1
1.1.1. Características físicas y componentes principales de los <i>Cuadrirrotores</i>	3
1.2. Antecedentes y trabajos previos	5
1.3. Objetivos	7
1.4. Contribuciones	7
2. Ecuaciones de movimiento para N vehículos tipo cuadrirrotor	9

2.1. Modelo matemático	9
3. Guiado, Navegación y Control de N agentes	15
3.1. Navegación distribuida para N agentes	15
3.1.1. Entradas de control virtual para N agentes	15
3.1.2. Marco de guiado en espacio euclidiano $SE(3)$	17
3.1.3. Control para la dinámica de posición de N agentes	18
3.1.4. Control para la dinámica de orientación de N agentes	19
4. Vuelo en Formación	22
4.1. Reglas de Reynolds	22
4.2. Teoría de Grafos	23
4.3. Matrices de Grafos	26
4.3.1. Matriz de Adyacencia o Conectividad	26
4.3.2. Matriz Laplaciana	27
4.4. Algoritmo de <i>Consensus</i>	27
4.4.1. <i>Consensus</i> en dinámicas de segundo orden	28
4.4.2. Algoritmo <i>Consensus no-lineal</i> por medio de control <i>Super Twisting</i>	30
5. Plataforma experimental	34
5.1. Cuadrirotor <i>Parrot</i> [®] <i>AR. Drone 2.0</i> [®]	34
5.2. Sistema de captura de movimiento <i>VICON</i> [®]	36

5.2.1. Hardware, Cámaras <i>VICON</i> ® <i>T-40</i> y <i>GIGANET</i>	36
5.2.2. Software de adquisición de datos <i>VICON Tracker</i> ®	38
5.3. Plataforma de desarrollo <i>LabVIEW</i> ®	39
5.4. Arquitectura del entorno de trabajo	40
6. Implementación en tiempo real	43
6.1. Pruebas del control <i>Super Twisting</i>	44
6.1.1. Trayectoria circular	45
6.1.2. Trayectoria en espiral	49
6.1.3. Respuesta a perturbaciones físicas	54
6.2. <i>Consensus</i> de segundo orden	58
6.2.1. Trayectoria circular con dos agentes (con perturbación)	61
6.2.2. Trayectoria para cubrir un área cuadrada	69
6.3. Propuesta de <i>Consensus no-lineal</i> en ley de control <i>Super Twisting</i> .	76
6.3.1. Trayectoria circular con dos agentes (con perturbación)	77
6.3.2. Trayectoria para cubrir un área cuadrada	83
6.3.3. Trayectoria en forma de lemniscata con tres agentes en perse- cución	89
7. Sumario, Conclusiones y Recomendaciones	98
7.1. Sumario	98
7.2. Conclusiones y discusión de resultados	99

7.3. Recomendaciones para trabajo futuro	100
A. Control por modos deslizantes	102
A.1. Modos Deslizantes para Sistemas de Segundo Orden	103
A.2. Eliminación del fenómeno de <i>castaño</i>	108
A.3. Técnica de control <i>Super Twisting</i>	109
B. Hipervínculos de los videos de la experimentación	113
B.1. Pruebas del control <i>Super Twisting</i>	113
B.2. <i>Consensus</i> de segundo orden	114
B.3. Propuesta de <i>Consensus no-lineal</i> en ley de control <i>Super Twisting</i> .	114

Índice de figuras

1.1. Diferentes tipos de <i>VANTs</i> dada su forma de sustentación.	3
1.2. Algunos componentes típicos de los <i>VANTs</i>	5
2.1. Marcos de referencia del cuadrirrotor.	10
4.1. Grafo básico de conexión entre 4 agentes.	24
4.2. Representación del flujo de información de i hacia j , notese que la comunicación es unidireccional.	24
4.3. Representación de un Grafo Dirigido.	27
5.1. Cuadrirrotor utilizado para la realización de experimentos <i>AR. Drone</i> <i>2.0</i> [®]	35
5.2. Cámara <i>VICON</i> [®] <i>T-40</i>	36
5.3. Elementos reflejantes que las cámaras del sistema siguen	36
5.4. Conmutador ethernet modelo <i>MX GIGANET</i>	37
5.5. Software de adquisición de datos <i>VICON Tracker</i> [®]	38
5.6. Visualización 3D del entorno capturado en el software <i>VICON Tracker</i> [®]	39
5.7. Ejemplo de interfaz utilizada en el software de desarrollo.	40

5.8. Arquitectura esquemática del entorno de trabajo.	42
6.1. Ángulos de guiñada con respecto al sistema de coordenadas del laboratorio	44
6.2. Trayectoria circular en el espacio tridimensional.	46
6.3. Ángulo de guiñada del cuadrirrotor siguiendo el círculo.	46
6.4. Gráficas de posiciones XYZ por eje para la trayectoria circular. . . .	47
6.5. Superficies deslizantes generadas utilizadas para el control en la trayectoria circular.	48
6.6. Señales de control del cuadrirrotor para la trayectoria circular.	48
6.7. Trayectoria en espiral en el espacio tridimensional.	51
6.8. Ángulo de guiñada del cuadrirrotor siguiendo el espiral.	51
6.9. Gráficas de posiciones XYZ por eje para la trayectoria en espiral. . .	52
6.10. Superficies deslizantes generadas utilizadas para el control en la trayectoria en espiral.	53
6.11. Señales de control del cuadrirrotor para la trayectoria en espiral. . . .	53
6.12. Cuadrirrotor en <i>vuelo estacionario</i>	54
6.13. Cuadrirrotor después de la perturbación.	55
6.14. Datos de posición durante el experimento de perturbación.	55
6.15. Velocidad en el eje Y debido a la perturbación inducida.	56
6.16. Ángulos de alabeo y cabeceo del cuadrirrotor durante la perturbación.	57
6.17. Superficies deslizantes producidas durante la perturbación.	57

6.18. Señales de control producidas durante la perturbación.	58
6.19. Arquitectura del sistema de navegación por <i>Consensus</i> de segundo orden.	60
6.20. Grafo de conexión utilizado para los experimentos de <i>Consensus</i> de segundo orden	61
6.21. Trayectoria circular con <i>Consensus</i> de segundo orden en el espacio tridimensional.	62
6.22. Ángulo de guiñada para el Agente 1.	62
6.23. Ángulo de guiñada para el Agente 2.	63
6.24. Posiciones en el eje X para el Agente 1.	63
6.25. Posiciones en el eje Y para el Agente 1.	64
6.26. Posiciones en el eje Z para el Agente 1.	64
6.27. Posiciones en el eje X para el Agente 2.	65
6.28. Posiciones en el eje Y para el Agente 2.	65
6.29. Posiciones en el eje Z para el Agente 2.	65
6.30. Señales generadas por el <i>Gestor Consensus</i> para el eje X	66
6.31. Señales generadas por el <i>Gestor Consensus</i> para el eje Y	66
6.32. Superficies deslizantes del Agente 1.	67
6.33. Superficies deslizantes del Agente 2.	67
6.34. Señales de control para el Agente 1.	68
6.35. Señales de control para el Agente 2.	68

6.36. Trayectoria deseada para cubrir un área cuadrada.	69
6.37. Seguimiento real para cubrir un área cuadrada.	70
6.38. Angulo de guiñada del Agente 1.	70
6.39. Angulo de guiñada del Agente 2.	70
6.40. Gráficas de posiciones <i>XYZ</i> por eje para la trayectoria para cubrir un área cuadrada del Agente 1.	71
6.41. Gráficas de posiciones <i>XYZ</i> por eje para la trayectoria para cubrir un área cuadrada del Agente 2.	72
6.42. Señales generadas por el <i>Gestor Consensus</i> para el eje X.	73
6.43. Señales generadas por el <i>Gestor Consensus</i> para el eje Y.	73
6.44. Superficies deslizantes del Agente 1.	74
6.45. Superficies deslizantes del Agente 2.	74
6.46. Señales de control para el Agente 1.	75
6.47. Señales de control para el Agente 2.	75
6.48. Arquitectura del sistema de coordinación utilizando solo controladores locales.	76
6.49. Trayectoria circular con <i>Consensus no-lineal</i> en el espacio tridimen- sional.	77
6.50. Ángulo de guiñada para el Agente 1.	78
6.51. Ángulo de guiñada para el Agente 2.	78
6.52. Gráficas de posiciones <i>XYZ</i> por eje para la trayectoria circular con <i>Consensus no-lineal</i> del Agente 1.	79

6.53. Gráficas de posiciones XYZ por eje para la trayectoria circular con <i>Consensus no-lineal</i> del Agente 2.	80
6.54. Superficies deslizantes para el Agente 1.	81
6.55. Superficies deslizantes para el Agente 2.	81
6.56. Señales de control para el Agente 1.	82
6.57. Señales de control para el Agente 2.	82
6.58. Seguimiento real para cubrir un área cuadrada.	83
6.59. Ángulo de guiñada para el Agente 1.	84
6.60. Ángulo de guiñada para el Agente 2.	84
6.61. Gráficas de posiciones XYZ por eje para la trayectoria para cubrir un área cuadrada con <i>Consensus no-lineal</i> del Agente 1.	85
6.62. Gráficas de posiciones XYZ por eje para la trayectoria para cubrir un área cuadrada con <i>Consensus no-lineal</i> del Agente 2.	86
6.63. Superficies deslizantes para el Agente 1.	87
6.64. Superficies deslizantes para el Agente 2.	87
6.65. Señales de control para el Agente 1.	88
6.66. Señales de control para el Agente 2.	88
6.67. Lemniscata.	89
6.68. Trayectoria descrita por los tres agentes realizando una lemniscata. .	90
6.69. Ángulo de guiñada del Agente 1.	91
6.70. Ángulo de guiñada del Agente 2.	91

6.71. Ángulo de guiñada del Agente 3	91
6.72. Gráficas de posiciones <i>XYZ</i> por eje para la trayectoria en forma de <i>Lemniscata</i> con <i>Consensus no-lineal</i> del Agente 1.	92
6.73. Gráficas de posiciones <i>XYZ</i> por eje para la trayectoria en forma de <i>Lemniscata</i> con <i>Consensus no-lineal</i> del Agente 2.	93
6.74. Gráficas de posiciones <i>XYZ</i> por eje para la trayectoria en forma de <i>Lemniscata</i> con <i>Consensus no-lineal</i> del Agente 3.	94
6.75. Superficies deslizantes para los 3 agentes realizando una <i>Lemniscata</i> . .	95
6.76. Señales de control para los 3 agentes realizando una <i>Lemniscata</i> . . .	96
 A.1. Superficie deslizante en el plano de fase.	103
A.2. Diagrama de bloques en el entorno de <i>Simulink</i>	106
A.3. Salida de desplazamiento.	106
A.4. Señal de control conmutando entre valores positivos y negativos. . .	107
A.5. Superficie Deslizante.	107
A.6. Acercamiento de la zona donde se parecencia el efecto del <i>castaño</i> . .	108
A.7. Salida de desplazamiento con el controlador <i>Super Twisting</i>	111
A.8. Señal de control producida por la técnica de <i>Super Twisting</i>	111
A.9. Superficie deslizante producida por la ley de control <i>Super Twisting</i> . .	112

Índice de tablas

5.1. Especificaciones técnicas de las cámaras <i>T-40</i>	37
A.1. Valores de la simulación del sistema de segundo orden.	105
A.2. Valores de la simulación del sistema de segundo orden utilizando <i>Super</i> <i>Twisting</i>	110

Nomenclatura

$(.)^\vee$	Operador inverso a $(\hat{\cdot})$.
α_n	Ángulo de ataque.
β_n	Ángulo de derrape.
\dot{e}_c	Error de seguimiento de velocidad para la coordinación.
\dot{e}_{ξ_n}	Error de seguimiento de velocidad lineal.
$\dot{h}_i(t)$	Término de compensación
ϵ	Constante de compensación para <i>saturación</i> y <i>tangente hiperbólica</i> .
$(\hat{\cdot})$	Operador que cumple que $a \times b = \hat{a}b$.
$\hat{\Omega}$	Matriz <i>anti-simétrica</i> de velocidades angulares en el marco del cuerpo.
λ_i	Vector de estados del i – <i>esimo</i> agente.
\mathcal{G}_n	Marco de guiado en el espacio euclidiano $SE(3)$.
\mathcal{L}_n	Momento producido por el movimiento de alabeo.
\mathcal{M}_n	Momento producido por el movimiento de cabeceo.
\mathcal{N}_n	Momento producido por el movimiento de guiñada.
ω_k	Velocidad angular del rotor k .
Ω_n	Velocidad angular en el marco del cuerpo.
ϕ	Ángulo de alabeo.
ψ	Ángulo de guiñada.
ρ_a	Densidad del aire.
$\rho_p(t, \sigma_{p_n})$	Agrupación de términos de la dinámica de posición.
σ	Superficie deslizando.
σ_c	Término de coordinación de la superficie deslizando (Superficie deslizando de coordinación).

σ_g	Superficie deslizante general.
σ_{a_n}	Superficie deslizante para el control de orientación.
σ_{p_n}	Superficie deslizante para el control de posición.
τ_a	Momentos producidos por los rotores sobre el cuerpo del cuadirrotor.
τ_{ϕ_n}	Momento producido en el eje X del cuadirrotor.
τ_{ψ_n}	Momento producido en el eje Z del cuadirrotor.
τ_{θ_n}	Momento producido en el eje Y del cuadirrotor.
θ	Ángulo de cabeceo.
$\tilde{\zeta}_i$	Vector de errores de salida para el agente i .
$\tilde{\zeta}_j$	Vector de errores de salida para el agente j .
$\Upsilon(R_n R_{d_n})$	Distancia entre las matrices de rotación real y deseada.
ξ_d	Referencias deseadas de posición.
ξ_n	Vector de posición en el marco inercial.
ζ	Vector de salida medida del i – <i>esimo</i> agente.
A	Matriz de adyacencia o conectividad.
a	Distancia focal en la trayectoria de lemniscata.
A_{d_k}	Área del disco formado por el rotor k .
a_{ij}	Peso de la arista de conexión desde j hasta i .
b	Ganancia del término de integración del control <i>Super Twisting</i> .
b_g	Vector binormal de control.
B_n	Marco de referencia del cuerpo.
B_{a_n}	Matriz diagonal definida positiva con ganancias b .
B_{p_n}	Matriz diagonal definida positiva con ganancias b .
c_n	Ganancia del control <i>Super Twisting</i> .
C_{a_n}	Matriz diagonal definida positiva con ganancias c_n .
C_{p_n}	Matriz diagonal definida positiva con ganancias c_n .
C_{Q_k}	Coefficiente de momento para el rotor k .
C_{T_k}	Coefficiente de empuje del rotor k .
D	Matriz diagonal de grados de entrada.
d_n	Distancia desde el centro de gravedad del cuadirrotor hasta el eje de giro de

	un rotor.
d_R	Perturbaciones giroscópicas e inerciales.
D_{η_n}	Momentos giroscópicos y aerodinámicos tomados como perturbaciones.
$d_{\eta_{aeron}}$	Momentos aerodinámicos.
$d_{\eta_{gyron}}$	Momentos giroscópicos.
D_{ξ_n}	Perturbaciones aerodinámicas.
d_{ξ_n}	Perturbaciones aerodinámicas.
D_{a_n}	Resistencia aerodinámica al avance.
E	Serie de aristas de conexión.
e	Error de seguimiento.
e_1	Vector de base canónica \mathbb{R}^3 definido como $e_1 = [1\ 0\ 0]^\top$.
e_2	Vector de base canónica \mathbb{R}^3 definido como $e_2 = [0\ 1\ 0]^\top$.
e_3	Vector de base canónica \mathbb{R}^3 definido como $e_3 = [0\ 0\ 1]^\top$.
e_Ω	Error de seguimiento de velocidad angular.
e_c	Error de seguimiento de posición para la coordinación.
e_e	Factor de alargamiento del radio mayor de la elipse.
e_R	Error de seguimiento de posición angular.
e_{ξ_n}	Error de seguimiento de posición.
f	Revoluciones por segundo de la circunferencia.
f_e	Revoluciones por segundo de la elipse formada en la trayectoria en espiral.
f_g	Vector de dirección hacia adelante (forward vector).
f_k	Fuerza generada sobre el cuadrirrotor por el rotor k .
f_{lm}	Revoluciones por segundo de la lemniscata.
G	Grafo con nodos y aristas de conexión.
g	Constante de la fuerza de gravedad.
h_i	Vector de posiciones deseadas de la formación.
H_n	Ganancia constante positiva del control por modos deslizantes.
h_{fe}	Altura final del espiral.
h_{ie}	Altura inicial del espiral.
I_n	Marco de referencia inercial.

$I_{r_{kn}}$	Momento de inercia para el rotor k .
J_n	Matriz que describe los momentos de inercia del $n - \text{esimo}$ cuadrirrotor.
k	Indicador del número del rotor utilizado por el cuadrirrotor.
$K = [k_1 \ k_2]$	Ganancias para mantener la posición y velocidad en lazo cerrado.
k_0	Constante positiva de la superficie deslizante.
$K_f = [k_{f_g} \ k_{f_g}]$	Ganancias que garantizan que los agentes lleguen al <i>consensus</i> .
$K_g = [k_{p_g} \ k_{v_g}]$	Ganancias que garantizan la convergencia de los estados del agente a la referencia deseada.
k_p	Ganancia del protocolo <i>consensus</i> para igualar posición.
k_v	Ganancia del protocolo <i>consensus</i> para igualar velocidad.
L	Matriz Laplaciana.
L_{a_n}	Fuerza de sustentación.
m_n	Masa del cuadrirrotor.
n_g	Vector normal de control.
N_i	Conjunto de i nodos.
n_n	Vector normal real.
p	Velocidad angular en el eje X del marco del cuerpo.
q	Velocidad angular en el eje Y del marco del cuerpo.
Q_k	Momento reactivo producido por el rotor k .
r	Radio de la circunferencia.
r	Velocidad angular en el eje Z del marco del cuerpo.
R_d	Matriz de rotación deseada.
r_k	Radio del rotor k .
R_n	Matriz de rotación del marco del cuerpo al marco inercial.
t	Tiempo.
t_f	Tiempo en alcanzar la altura final del espiral.
T_k	Fuerza de empuje del rotor k del cuadrirrotor.
T_T	Fuerza de empuje total de los rotores del cuadrirrotor.
u_i	Salida de control <i>Super Twisting</i> con coordinación por <i>Consensus</i> .
u_{a_n}	Entrada de control virtual para la dinámica de orientación.

u_{p_n}	Entrada de control virtual para la dinámica de posición.
V	Serie de N nodos.
v_i	Nodo i .
v_j	Nodo j .
V_n	Velocidad lineal en el marco inercial.
w	Término integral del control <i>Super Twisting</i> .
W_n	Marco de referencia del viento.
W_n	Matriz de rotación del marco del cuerpo al marco del viento.
x	Coordenada de posición en el eje X para la trayectoria circular.
x_e	Coordenada de posición en el eje X para la trayectoria en espiral.
x_G	Vector de estados global (para todos los agentes).
$x_{centroide}$	Coordenada de posición del centro de la circunferencia en el eje X para la trayectoria circular.
x_{lm}	Coordenada de posición en el eje X para la trayectoria de lemniscata.
x_{ref}	Referencia deseada.
$x_{centroide_e}$	Coordenada de posición del centro de la circunferencia en el eje X para la trayectoria en espiral.
y	Coordenada de posición en el eje Y para la trayectoria circular.
y_e	Coordenada de posición en el eje Y para la trayectoria en espiral.
Y_{a_n}	Fuerza aerodinámica lateral.
$y_{centroide}$	Coordenada de posición del centro de la circunferencia en el eje Y para la trayectoria circular.
y_{lm}	Coordenada de posición en el eje Y para la trayectoria de lemniscata.
$y_{centroide_e}$	Coordenada de posición del centro de la circunferencia en el eje Y para la trayectoria en espiral.
z_e	Coordenada de posición en el eje Z para la trayectoria en espiral.
$VANT$	Vehículo aéreo no tripulado.
sat	Función matemática <i>saturación</i> .
sgn	Función matemática <i>signo</i> .
$\text{Skew}(A)$	Matriz anti-simétrica de A .

Agradecimientos

A mis padres, Gilberto y María de la Luz, por ser mi ejemplo y modelo a seguir; que siempre me brindan su completo y sincero apoyo en cada una de mis metas, ya que siempre están ahí cuando los necesito, e innumerables veces han sabido como guiarme de la manera más conveniente.

De una manera especial, a mi hermano, porque a pesar de haber recorrido menos camino, siempre tiene una solución para todo y busca apoyarme de alguna forma.

Al Centro de Investigación e Innovación en Ingeniería Aeronáutica por haberme recibido con las puertas abiertas, incluso desde antes de iniciar mis estudios de posgrado.

A mi *alma mater*, la Facultad de Ingeniería Mecánica y Eléctrica, por haberme provisto de mi formación profesional y haber sido mi segunda casa durante los últimos años.

A la Universidad Autónoma de Nuevo León, que siendo la institución madre de mis estudios, sentó las bases y estatutos para mi formación.

Al Consejo Nacional de Ciencia y Tecnología por el apoyo económico para la realización de mis estudios de posgrado.

También agradezco a aquellas personas que de alguna u otra manera me brindaron su apoyo para poder llegar a la culminación de este proyecto, como los son la Dra. Patricia Zambrano, Gabino Ramírez, Daniel Pérez Banda y todos aquellos que

estuvieron a lo largo del camino.

A mi comité de tesis conformado por el M.C. Daniel Librado Martínez y el Dr. Anand Eleazar Sánchez Orta, por haberse tomado el tiempo de revisar este trabajo.

A Delia Saucedo, que siempre me apoyó durante este tiempo y, que cuando necesitaba a alguien en quien confiar, estaba ahí.

Y finalmente, de una forma muy especial, a aquella persona, que durante el inicio de este posgrado, me dio una oportunidad, que creyó en mi a pesar de las dificultades, que transmitió su conocimiento confiando en mi capacidad, y que me defendió de los obstáculos que surgieron durante este tiempo. A mi asesor, el Dr. Octavio García Salazar.

Resumen

El vuelo en formación es una de las misiones que un conjunto de vehículos realiza para llegar a un objetivo común, que por lo general, es necesario llevar a cabo de una manera eficiente, rápida y más segura que con el uso de un solo vehículo. Es por ello que este campo de estudio ha ganado gran interés por parte de la comunidad científica, buscando solventar las necesidades que este tipo de aplicaciones implican.

La principal problemática que se presenta para lograr el vuelo en formación radica en la necesidad de coordinar los vehículos que lo conforman, debido a que es necesario que mantengan la geometría y velocidad de la formación, independientemente de los factores externos. Además, deben de ser capaces de tomar decisiones en caso de que alguno de los vehículos cambie su ruta, para que el resto del grupo se ajuste a las nuevas condiciones de vuelo.

En este trabajo de investigación se busca resolver esta problemática por medio del diseño e implementación de algoritmos que logren el vuelo en formación, utilizando la técnica de *consensus*. Para lograr este objetivo, se analizan las ecuaciones de movimiento para N vehículos tipo cuádrirrotor, y se desarrollaron estrategias de *Guiado, Navegación y Control*.

Posteriormente, se tratan los conceptos generales del vuelo en formación y se proponen dos algoritmos para llegar al *consensus*. El primero, de naturaleza de segundo orden, actuó como *Gestor* para los controles locales de cada vehículo, generando las referencias de posición y velocidad, en base a las trayectorias definidas. El segundo algoritmo propuesto, establecido como *consensus no-lineal* basado en la ley de control de *Super Twisting*, utiliza solo los controladores locales de posición,

con intercambio de información entre ellos, para realzar la coordinación.

Finalmente, para poner a prueba estos algoritmos, se realiza una serie de experimentos que permiten evaluar y analizar el desempeño de ambos, con lo cual es posible establecer el comportamiento y características de cada una de estas aproximaciones para el vuelo en formación.

Capítulo 1

Introducción

En el presente capítulo se presenta la información de los *Vehículos Aéreos No Tripulados*, su clasificación, características principales y sus componentes de operación. Los trabajos previos relacionados al *Vuelo en Formación* serán presentados así como los objetivos y contribuciones realizadas en este trabajo de investigación.

1.1 Vehículos Aéreos No Tripulados

Un *vehículo aéreo no tripulado* (abreviado como VANT o UAV¹) es un tipo de aeronave cuya principal característica es que puede realizar su misión sin tener la operación directa de un humano, teniendo la capacidad de funcionar de manera autónoma, pre-programada, o controlada de forma remota. Es por esto que este tipo de aeronave presenta gran flexibilidad y variedad de aplicaciones militares y civiles debido a su capacidad de poder mantener vuelo a bajas velocidades y muy cerca del suelo, así como una gran capacidad de maniobra. Entre sus principales aplicaciones se tienen la vigilancia, patrullaje, persecución, reconocimiento en áreas peligrosas, así como agricultura de precisión, mapeo y adquisición de datos geográficos, monitoreo de condiciones climatológicas, entre otras.

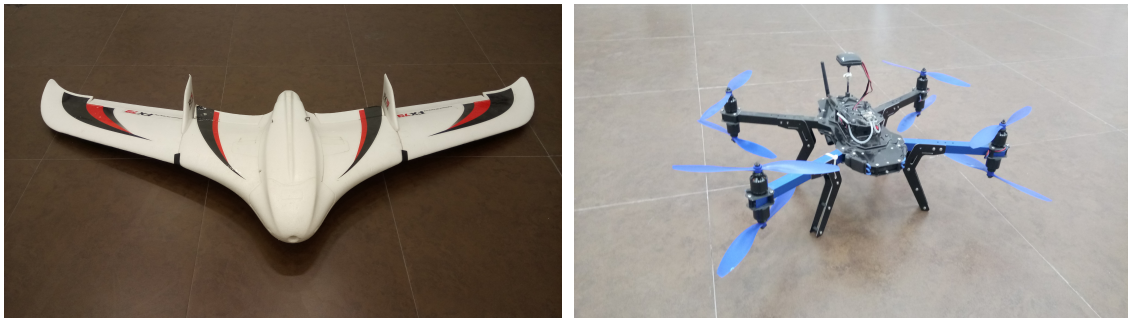
¹Por las siglas en inglés de *Unmanned Aerial Vehicle*

Existe gran variedad y tipos de VANT, sin embargo la clasificación más general que puede realizarse es en base a su forma de sustentación, es decir, aquellos que son de *ala fija* o de *ala rotativa*.

- **Ala fija:** son aquellas aeronaves cuya configuración principal consiste en un fuselaje con un ala acoplada la cual es la principal fuente de sustentación. Junto a lo anterior, se tiene una fuente generadora de fuerza de empuje, es decir, algún tipo de motor. Este tipo de aeronaves realizan su despegue y aterrizaje de manera horizontal lo que les obliga a necesitar de una pista para alcanzar la velocidad suficiente para comenzar el vuelo. En cuestiones de maniobrabilidad, utilizan superficies de control acopladas al ala y a las distintas superficies sustentadoras. Lo anterior para crear diferencias en los momentos de inercia generados por las distintas partes de la aeronave y así realizar los giros y movimientos de la misma, ver Figura 1.1a.
- **Ala rotativa:** este tipo de aeronaves generan sustentación haciendo girar una hélice o conjunto de hélices las cuales tienen las características aerodinámicas para lograr tal objetivo. En el caso de los *multirotores*, se tiene más de un solo conjunto de motor-hélice. Dado lo anterior, se posee mayor estabilidad y controlabilidad de la aeronave. Además, a diferencia de las de ala fija, no es necesario tener superficies de control adicionales para realizar los movimientos y giros necesarios, dado que con la correcta manipulación de los distintos rotores es posible lograrlos. Junto con estas características, los *multirotores* se destacan por su redundancia mecánica, dado que al poseer varios conjuntos de motores y palas, en caso de daño o falla de alguno de ellos, es posible compensar su funcionamiento con los restantes, ver Figura 1.1b.

Establecida la clasificación anterior y para los fines de este trabajo, se decidió utilizar los VANT del tipo *ala-rotativa* y específicamente *multi-rotores* de cuatro conjuntos de *motor-pala*, debido a las ventajas que presentan sobre aquellos de *ala*

fija, además de ser comercialmente más accesibles. En lo restante de este trabajo, se procede a referirse solamente a los **cuadrirrotores** como objeto de estudio.



(a) Ala fija.

(b) Ala rotativa.

Figura 1.1: Diferentes tipos de VANTs dada su forma de sustentación.

1.1.1 Características físicas y componentes principales de los *Cuadrirrotores*

Los VANT, de *ala fija* o *ala rotativa*, tienen características físicas, mecánicas y electrónicas, entre las cuales se pueden mencionar:

- **Computadora de vuelo:** está constituida por los distintos procesadores de datos necesarios para tomar decisiones y acciones de control en base a la información proveída por los sensores de la aeronave. Básicamente se encarga de garantizar el correcto control y maniobras del vehículo así como traducir las distintas acciones a nivel *hardware* en base a las ordenes pre-programadas o dadas por un operador, ver Figura 1.2a.
- **Sensores de orientación y posición:** se encargan de adquirir los datos necesarios para que la computadora de vuelo pueda calcular los valores de orientación, posición y demás datos necesarios para el vehículo.

- **Motores eléctricos y controladores electrónicos de velocidad:**
 - *Motores:* los motores son los encargados de darle propulsión a la aeronave. En el caso de los cuadrirrotores se tienen cuatro motores los cuales, en conjunto con sus palas, son los encargados de propulsar el vehículo así como de dotarle de los movimientos necesarios para realizar las maniobras pertinentes por medio de la combinación del funcionamiento de los mismos, ver Figura 1.2c.
 - *Controladores electrónicos de velocidad:* para poder manipular el funcionamiento de los motores eléctricos es necesario convertir las ordenes de la computadora de vuelo a señales eléctricas que los motores puedan utilizar, es por eso que se utilizan este tipo de controladores. Por lo general, la computadora de vuelo envía señales proporcionales a la velocidad deseada del motor por medio de *modulación de ancho de pulsos* o *PWM*² para posteriormente mandar las señales de voltaje y potencia adecuadas a los motores, ver Figura 1.2d.
- **Actuadores:** en el caso de que el vehículo tenga una tarea específica la cual involucre manipular o interactuar con el entorno físico que le rodea, es muy posible que cuente con actuadores. Estos son elementos, ya sea electrónicos o mecánicos, los cuales les permiten realizar acciones en el entorno, tales como tomar objetos, moverlos, modificar mecanismos, y realizar distintas tareas.
- **Estructura o cuerpo del vehículo:** como su nombre lo dice, es la principal estructura del vehículo la cual se encarga de darle soporte a todos los elementos del mismo. En el caso de los *cuadrirrotores*, el diseño más básico consta de bastidores en forma de cruz o **X**, lo que le brinda la capacidad de poseer motores en cada uno de los extremos formados. Además, se tiene el conjunto de los demás componentes distribuido a lo largo de dicha estructura procurando siempre mantener el centro de gravedad de todo el vehículo lo más cerca de su centro, ver Figura 1.2b.

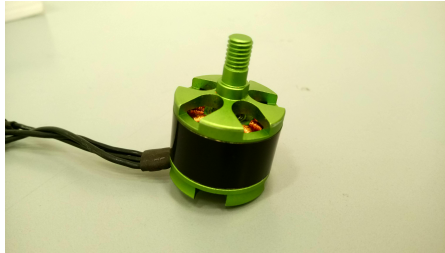
²De las siglas en inglés para Pulse Width Modulation



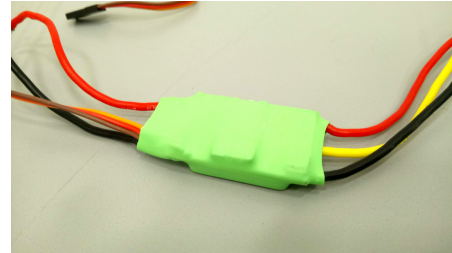
(a) Computadora de vuelo.



(b) Partes estructurales de un cuadrirrotor.



(c) Motor eléctrico sin escobillas.



(d) Control electrónico de velocidad (ESC).

Figura 1.2: Algunos componentes típicos de los *VANTs*.

1.2 Antecedentes y trabajos previos

El vuelo en formación es una de las características de los sistemas de vehículos *multi-agentes* el cual es inspirado por el comportamiento colaborativo de seres vivos, como manadas de animales, parvadas de pajaros, bancos de peces, colonias de bacterias, entre otras [1]. Este tipo de comportamiento otorga distintas ventajas tales como un incremento en la rapidez de la realización de distintas tareas, mayor eficiencia en la captura de datos, tolerancia a fallos debido a la redundancia y presencia de múltiples vehículos trabajando en tareas independientes [2], [3], [4] .

Como resultado de las ventajas que presenta, así como sus potenciales aplicaciones, el vuelo en formación a sido estudiado desde diferentes perspectivas y estructuras. El problema más general es poder hacer que los agentes del sistema lleguen a una coordinación de sus parámetros de operación dadas las condiciones del vuelo, por lo que siempre se busca llegar a un «acuerdo» entre ellos en términos

de sus estados, por ejemplo de posición y velocidad. Lo anterior es conocido como el problema *Consensus* [5], [6], [7]. Distintas propuestas y aproximaciones han sido realizadas, utilizando varias técnicas y modelos ya aplicados [8], [9]; además, otros campos estudiados con aplicaciones distintas han sido adaptados a la cooperación, como lo es la *teoría de juegos*. En [10] y [11] se presentan avances en los algoritmos de coordinación utilizando esta herramienta. En [12] también se presenta una aproximación de coordinación de agentes por medio de la *teoría de juegos de suma cero*, en dicho trabajo es posible coordinar un conjunto de agentes dotando al líder de movimiento libre para que de esta manera los seguidores busquen seguir e imitar los movimientos del mismo.

De manera similar, otros conceptos como la *teoría de decisión* han sido estudiados para la aplicación en problemas de coordinación. Utilizando los preceptos de dicho campo, se analizan una serie de opciones que pueden ser tomadas cuando no se tiene certeza del resultado al que llevará la realización de alguna acción o cambio. Este concepto radica en la toma de la mejor decisión posible así como el establecimiento de lo que se considera la «mejor» decisión dependiendo del estado del sistema. Esta teoría constituye una gran herramienta para la toma de decisiones en escenarios impredecibles y con condiciones de operación variantes como se establece en [13].

Otra aproximación es el modelado de pares de agentes en términos de las distancias que hay entre ellos. De esta manera se utilizan algoritmos que seleccionen al líder adecuado para la formación y buscar una solución óptima para mantener la distancia entre estos agentes, y si es necesario o requerido, poder cambiar fácilmente los roles entre agentes y seguidores. En [14] se proponen soluciones para este tipo de estructura y se presentan resultados, y en [15] se exponen algoritmos adaptativos con topologías cambiantes.

1.3 Objetivos

El objetivo principal de este trabajo de investigación es el diseño e implementación en tiempo real, de un algoritmo que sea capaz de coordinar vehículos tipo *cuadrirrotor*, para realizar vuelo en formación. Para llegar a este objetivo, es necesaria la realización de una serie de tareas específicas las cuales pueden dividirse como:

- Diseño de un control por *modos deslizantes*, en su variante de *Super Twisting*, para controlar las dinámicas de orientación y posición de los cuadrirrotores, tomando en cuenta un marco de guiado en el espacio euclidiano $SE(3)$.
- Implementación en tiempo real del controlador desarrollado, evaluando su comportamiento y desempeño frente a los factores del vuelo en formación.
- Diseño de algoritmos para el vuelo en formación utilizando las *Reglas de Reynolds* y la metodología de *Consensus* como base.
- Implementación en tiempo real de los algoritmos para el vuelo en formación; y utilizando dos o más cuadrirrotores, realizar trayectorias definidas para evaluar el comportamiento y desempeño de los distintos algoritmos.

1.4 Contribuciones

Como parte de este trabajo de investigación, se realizan una serie de contribuciones científicas en base a los objetivos propuestos. Estas pueden dividirse principalmente en las siguientes:

- Diseño de un algoritmo de vuelo en formación basado en *Consensus* de segundo orden.

-
- Diseño de un algoritmo de *Consensus no-lineal* basado en el control *Super Twisting* para la navegación distribuida de N agentes.
 - Implementación en tiempo real de los algoritmos de *Consensus* para el vuelo en formación del sistema de *multi-agentes*.

Capítulo 2

Ecuaciones de movimiento para N vehículos tipo cuadrirrotor

Para la realización de este trabajo de investigación, se utilizan aeronaves del tipo cuadrirrotor con estructura en forma de \mathbf{X} , considerados como agentes de la formación. A continuación se presentan las ecuaciones de movimiento del VANT basado en el enfoque *Newton-Euler* [16].

2.1 Modelo matemático

Para la obtención de la posición y orientación de N cuadrirrotores; esto es $n = 1, \dots, N$, considerados agentes, dentro del espacio tridimensional, es necesario definir los marcos de referencia como sigue [16], [17], ver Figura 2.1.

- Inercial (Inertial frame ¹) $I_n = \{x_{I_n}, y_{I_n}, z_{I_n}\}$: Es el marco de referencia principal el cual es fijo y definido en «tierra».
- Cuerpo (Body frame ¹) $B_n = \{x_{B_n}, y_{B_n}, z_{B_n}\}$: Marco de referencia acoplado al centro de gravedad (*c.g.*) del vehículo aéreo.
- Viento (Wind-frame ¹) $W_n = \{x_{W_n}, y_{W_n}, z_{W_n}\}$: Marco de referencia considerado

¹Nombre correspondiente en inglés

durante el vuelo en crucero y es utilizado para modelar efectos aerodinámicos.

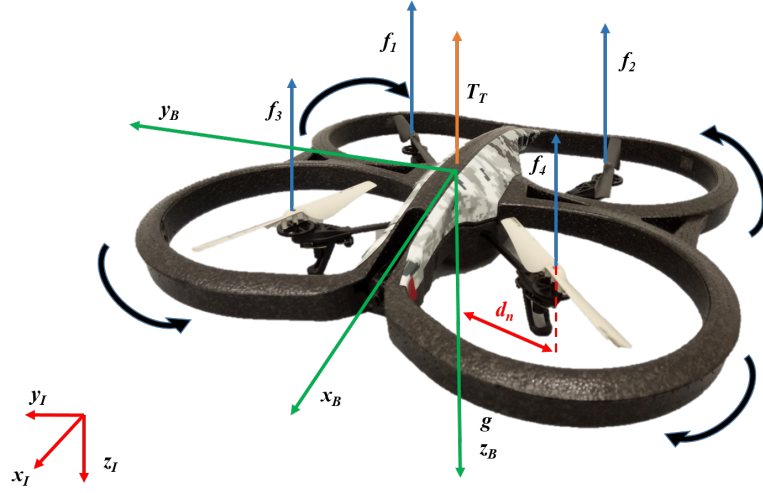


Figura 2.1: Marcos de referencia del cuadirrotor.

Por consiguiente, las ecuaciones de movimiento para N agentes son:

$$\dot{\xi}_n = V_n \quad (2.1)$$

$$m_n \dot{V}_n = R_n(-T_{T_n} e_3) + m_n g e_3 + D_{\xi_n} \quad (2.2)$$

$$\dot{R}_n = R_n \hat{\Omega}_n \quad (2.3)$$

$$J_n \dot{\Omega}_n = -\Omega_n \times J_n \Omega_n + \tau_{a_n} + D_{\eta_n} \quad (2.4)$$

de donde se pueden describir los siguientes términos:

- $\xi_n = (x_n, y_n, z_n)^\top \in \mathbb{R}^3$: Representa las coordenadas de la posición con respecto al marco de referencia inercial. Dado esto, se puede entender que V_n representa la velocidad lineal del agente en el marco inercial fijo.
- $R_n \in SO(3) : \mathcal{B}_n \rightarrow \mathcal{I}_n$: Es una matriz de rotación ortogonal, definida del marco del *Cuerpo* al marco *Inercial*, y está parametrizada por los ángulos de *Euler* ϕ_n , θ_n y ψ_n .

$$R_n = \begin{pmatrix} c_{\theta_n} c_{\psi_n} & s_{\phi_n} s_{\theta_n} c_{\psi_n} - c_{\phi_n} s_{\psi_n} & c_{\phi_n} s_{\theta_n} c_{\psi_n} + s_{\phi_n} s_{\psi_n} \\ c_{\theta_n} s_{\psi_n} & s_{\phi_n} s_{\theta_n} s_{\psi_n} + c_{\phi_n} c_{\psi_n} & c_{\phi_n} s_{\theta_n} s_{\psi_n} - s_{\phi_n} c_{\psi_n} \\ -s_{\theta_n} & s_{\phi_n} c_{\theta_n} & c_{\phi_n} c_{\theta_n} \end{pmatrix}$$

Las funciones trigonométricas se representan como $c_x = \cos(x)$ y $s_x = \sin(x)$ por cuestiones de simplificación de nomenclatura.

- $\Omega_n = (p_n, q_n, r_n)^\top \in \mathbb{R}^3$: Es la velocidad angular del agente con respecto al marco del cuerpo.
- $\hat{\Omega}_n$: Corresponde a la matriz anti-simétrica de la velocidad angular en el marco del cuerpo; esto es $a \times b = \hat{a}b$, lo cual resulta

$$\hat{\Omega}_n = \begin{pmatrix} 0 & -r_n & q_n \\ r_n & 0 & -p_n \\ -q_n & p_n & 0 \end{pmatrix}$$

- $T_{T_n} \in \mathbb{R}$: Fuerza de empuje total producida por los rotores del vehículo y puede ser caracterizada como:

$$T_{T_n} = \begin{pmatrix} 0 \\ 0 \\ \sum_{k=1}^4 T_k \end{pmatrix}$$

donde se define $T_k = \rho_a A_{d_k} \omega_k^2 r^2 C_{T_k}$ y a su vez ρ_a corresponde a la densidad del aire, A_{d_k} es el área del disco rotor², ω_k describe la velocidad angular de un rotor, r_k es el radio de un rotor³ y C_{T_k} es el coeficiente de empuje de un solo conjunto motor-hélice, es decir, para $k = 1, 2, 3, 4$.

- $\tau_a \in \mathbb{R}^3$: Momentos producidos por los propulsores del cuadirrotor sobre su centro de gravedad. Haciendo uso de estos momentos, es posible describir los

²Corresponde al área descrita por un círculo cuyo diámetro corresponde a la envergadura total, es decir, de punta a punta de la hélice en cuestión.

³De igual manera corresponde al radio descrito por el círculo formado por la envergadura de la hélice.

movimientos angulares del cuadrirrotor. Están definidos como:

$$\tau_{a_n} = \begin{pmatrix} \tau_{\phi_n} \\ \tau_{\theta_n} \\ \tau_{\psi_n} \end{pmatrix} = \begin{pmatrix} d_n [(f_{2_n} + f_{3_n}) - (f_{4_n} + f_{1_n})] \\ d_n [(f_{3_n} + f_{4_n}) - (f_{1_n} + f_{2_n})] \\ Q_{1_n} - Q_{2_n} + Q_{3_n} - Q_{4_n} \end{pmatrix}$$

donde d_n representa la distancia desde el centro de gravedad del agente al eje de giro del rotor (de manera axial), además f_{k_n} denota la fuerza de empuje generada por un solo propulsor o rotor, esto para $k = 1, 2, 3, 4$. $Q_k = \rho_a A_{d_k} \omega_k^2 r_k^3 c_{Q_k}$ corresponde al par o momento reactivo de los rotores con un c_{Q_k} que es el coeficiente de momento del eje de un rotor para $k = 1, 2, 3, 4$.

- e_1, e_2, e_3 : Corresponden a vectores de base canónica de \mathbb{R}^3 , definidos como $e_1 = [1\ 0\ 0]^\top$, $e_2 = [0\ 1\ 0]^\top$, $e_3 = [0\ 0\ 1]^\top$
- $m_n \in \mathbb{R}$: Corresponde a la masa del n -esimo agente.
- $J_n \in \mathbb{R}^3$: Matriz que describe los momentos de inercia del n -esimo agente.
- D_{ξ_n} : Representa las perturbaciones aerodinámicas y están definidas como:

$$D_{\xi_n} = \begin{pmatrix} d_{\xi_{1_n}} \\ d_{\xi_{2_n}} \\ d_{\xi_{3_n}} \end{pmatrix} = R_n W_n^T \begin{pmatrix} D_{a_n} \\ Y_{a_n} \\ L_{a_n} \end{pmatrix}$$

donde D_{a_n} , Y_{a_n} , y L_{a_n} corresponden a las fuerzas aerodinámicas generadas durante el vuelo, esto es, la *Resistencia al avance*, *Fuerza Lateral* y *Fuerza de Sustentación*, respectivamente. Además, también se utiliza una matriz de rotación $W_n : \mathcal{B}_n \rightarrow \mathcal{A}_n$ que orienta las fuerzas correspondientes del marco inercial del *Cuerpo* al del *Viento*. Está definida de la siguiente manera:

$$W_n = \begin{pmatrix} c_{\alpha_n} c_{\beta_n} & s_{\beta_n} & s_{\alpha_n} c_{\beta_n} \\ -c_{\alpha_n} s_{\beta_n} & c_{\beta_n} & -s_{\alpha_n} s_{\beta_n} \\ -s_{\alpha_n} & 0 & c_{\alpha_n} \end{pmatrix} \quad (2.5)$$

de donde α_n y β_n son el ángulo de *ataque* y de *derrape* de la N -esima aeronave, respectivamente, [18], [19], [20].

- D_{η_n} : Representa los momentos giroscópicos y aerodinámicos los cuales son tomados como perturbaciones. Estos son definidos como:

$$D_{\eta_n} = \begin{pmatrix} d_{\eta_{1n}} \\ d_{\eta_{2n}} \\ d_{\eta_{3n}} \end{pmatrix} = d_{\eta_{gyron}} + d_{\eta_{aeron}}$$

de donde los momentos giroscópicos se definen de tal manera que:

$$d_{\eta_{gyron}} = \sum_{k=1}^4 (-1)^{k+1} I_{r_k} [\Omega \times e_3 \omega_k]$$

y afectando todos los términos se obtiene:

$$d_{\eta_{gyron}} = \begin{pmatrix} q_n(I_{r_{1n}} \omega_{r_{1n}} - I_{r_{2n}} \omega_{r_{2n}} + I_{r_{3n}} \omega_{r_{3n}} - I_{r_{4n}} \omega_{r_{4n}}) \\ p_n(-I_{r_{1n}} \omega_{r_{1n}} + I_{r_{2n}} \omega_{r_{2n}} - I_{r_{3n}} \omega_{r_{3n}} + I_{r_{4n}} \omega_{r_{4n}}) \\ 0 \end{pmatrix}$$

utilizando $I_{r_{k_n}}$ que es el momento de inercia correspondiente al rotor k y ω_{k_n} correspondiendo a la velocidad angular de los mismos con $k=1, 2, 3, 4$.

De la misma manera, los momentos generados durante el vuelo en crucero se representan como:

$$d_{\eta_{aeron}} = \begin{pmatrix} \mathcal{L}_n \\ \mathcal{M}_n \\ \mathcal{N}_n \end{pmatrix}$$

donde \mathcal{L}_n , \mathcal{M}_n y \mathcal{N}_n son los momentos aerodinámicos correspondientes a los movimientos de alabeo, cabeceo y guiñada, respectivamente, [19], [16], [18].

COMENTARIO 2.1 Las ecuaciones (2.1) y (2.2) corresponden a la dinámica de posición o traslación del agente en el espacio eucladiano. Representan las ecuaciones de Navegación y Fuerzas

COMENTARIO 2.2 *La dinámica de orientación o rotación del agente es descrita por las ecuaciones (2.3) y (2.4) y representan las ecuaciones de Orientación y Momentos.*

COMENTARIO 2.3 *Es necesario notar que el empuje total T_{T_n} es multiplicado por el vector de base canónica e_3 debido a que solo actúa en el eje z_{B_n} del agente.*

Capítulo 3

Guiado, Navegación y Control de N agentes

En este capítulo se presentará el *Guiado, Navegación y Control* para N cuádrirrotores considerados como agentes. Esto se realiza por medio de los algoritmos de control para las dinámicas del *cuádrirrotor* descritas en el capítulo anterior, los cuales serán diseñados por medio de técnica de control por *Modos Deslizantes*.

3.1 Navegación distribuida para N agentes

Para que una formación pueda realizar las trayectorias requeridas o tareas pre-establecidas, es necesario que cada uno de los agentes obedezca las ordenes dadas por medio de un algoritmo de control. En efecto, se diseña en base al control *Modos deslizantes* con *Super Twisting* el algoritmo que garantizará que la navegación de los agentes se cumpla, ver el Apéndice A para más información referente a las bases del control.

3.1.1 Entradas de control virtual para N agentes

Retomando las ecuaciones de movimiento para N cuádrirrotores presentadas en

(2.1)-(2.4) como:

$$\begin{aligned}\dot{\xi}_n &= V_n \\ m_n \dot{V}_n &= R_n(-T_{T_n} e_3) + m_n g e_3 + D_{\xi_n} \\ \dot{R}_n &= R_n \hat{\Omega}_n \\ J_n \dot{\Omega}_n &= -\Omega_n \times J_n \Omega_n + \tau_{a_n} + D_{\eta_n}\end{aligned}$$

es posible definir la navegación de los agentes en el *espacio euclidiano* $SE(3)$ utilizando el control *Super Twisting*. Para lo anterior, se reescribe el modelo como:

$$\dot{\xi}_n = V_n \quad (3.1)$$

$$\dot{V}_n = u_{p_n} + d_{\xi_n} \quad (3.2)$$

$$\dot{R}_n = R_n \Omega_n \quad (3.3)$$

$$\dot{\Omega}_n = u_{a_n} + d_{R_n} \quad (3.4)$$

donde se puede definir;

- $u_{p_n} \in \mathbb{R}^3$: entrada de control *virtual* para la dinámica de *posición* y corresponde a los términos descritos como:

$$u_{p_n} = g e_3 + \frac{(R_n e_3) T_{T_n}}{m_n} \quad (3.5)$$

- $u_{a_n} \in \mathbb{R}^3$: entrada de control *virtual* para la dinámica de *orientación* cuyos términos corresponden a:

$$u_{a_n} = \frac{\tau_{a_n}}{J_n} \quad (3.6)$$

- $d_{\xi_n} = D_{\xi_n}$: perturbaciones debido a las fuerzas aerodinámicas, esto es, la acción de las mismas se considera como una perturbación para el sistema.
- $d_{R_n} = \Omega_n \times J_n \Omega_n + D_{\eta_n}$: perturbaciones debido efectos giroscópicos e inerciales. Se consideran estos términos como perturbaciones para el sistema para que simplificar su acción en el control.

3.1.2 Marco de guiado en espacio euclidiano $SE(3)$

Es necesario definir un marco de guiado para los agentes que considere los vectores de dirección para la navegación a lo largo de una trayectoria perteneciente al espacio euclidiano \mathbb{R}^3 . Dado esto, se define tal marco de guiado \mathcal{G}_n como:

$$\mathcal{G}_n = \{f_{g_n}, b_{g_n}, n_{g_n}\} \quad (3.7)$$

donde se tienen los siguientes vectores:

- n_{g_n} : vector normal de control. Este es definido en base a los errores de posición y velocidad tal que:

$$n_{g_n} = \frac{ge_3 + u_{p_n}}{\|ge_3 + u_{p_n}\|} \quad (3.8)$$

- f_{g_n} : vector de dirección hacia adelante¹. Es el vector unitario ortogonal al vector normal de control n_{g_n} por lo que se define como:

$$f_{g_n} = \frac{n_{g_n} \times e_1}{\|n_{g_n} \times e_1\|} \quad (3.9)$$

- b_{g_n} : vector binormal de control. Corresponde al vector ortogonal a ambos vectores descritos anteriormente y es definido como:

$$b_{g_n} = -(f_{g_n} \times n_{g_n}) \quad (3.10)$$

donde $n = 1, \dots, N$.

Una vez definidos estos vectores que corresponden al marco de guiado \mathcal{G}_n , es posible definir una *matriz de rotación deseada*, la cual se describe a continuación:

$$R_{d_n} = [f_{g_n} \quad b_{g_n} \quad n_{g_n}] \quad (3.11)$$

con $R_{d_n} \in SO(3)$.

¹Conocido en el idioma inglés como *forward vector*

3.1.3 Control para la dinámica de posición de N agentes

Para la realización del control de posición, se utilizan las ecuaciones (3.1) y (3.2) donde es posible definir una relación con el vector normal de control (3.8) esto resulta:

$$n_{g_n} = R_{d_n} e_3$$

donde se busca que la matriz de rotación deseada R_{d_n} sea igual a la real R_n , por lo que se puede establecer el vector *normal real* como $n_n = R_n e_3$. Por lo tanto se define una relación con la entrada virtual u_{p_n} como:

$$u_{p_n} = g e_3 + \frac{n_n^T T_{T_n}}{m_n} \quad (3.12)$$

Entonces, se procede al diseño del control utilizando el algoritmo de *Super Twisting*. Para el error de seguimiento, es necesario definir los errores correspondientes en base a los vectores de estados de posición y velocidad como:

$$\begin{aligned} e_{\xi_n} &= \xi_{d_n} - \xi_n \\ \dot{e}_{\xi_n} &= \dot{\xi}_{d_n} - \dot{\xi}_n \end{aligned} \quad (3.13)$$

Posteriormente, se diseña una superficie deslizante de manera similar a como se presenta en la ecuación (A.3) resultando de la siguiente manera:

$$\sigma_{p_n} = k_{p_n} e_{\xi_n} + \dot{e}_{\xi_n} \quad (3.14)$$

y utilizando las ecuaciones de la forma de (A.7) se diseña el control como:

$$\begin{aligned} u_{p_n} &= C_{p_n} \sqrt{|\sigma_{p_n}|} \operatorname{sgn}(\sigma_{p_n}) + w_{p_n} \\ \dot{w}_{p_n} &= B_{p_n} \operatorname{sgn}(\sigma_{p_n}) \end{aligned} \quad (3.15)$$

donde C_{p_n} y B_{p_n} son matrices diagonales definidas positivas cuyos elementos son calculados como $c_{p_n,v} = 1.5\sqrt{H_{p_n}}$ y $b_{p_n,v} = 1.1H_{p_n}$ de donde H_{p_n} es una ganancia positiva, ver Apéndice A, y con $v = 1, 2, 3$ correspondiendo a las distintas acciones de control para cada eje de movimiento.

3.1.3.1 Dinámica resultante de posición

La dinámica resultante del sistema se obtiene diferenciando la variable *deslizante* (3.14), de tal manera que se obtiene:

$$\dot{\sigma}_{p_n} = k_{p_n} \dot{e}_{\xi_n} + \ddot{e}_{\xi_n} \quad (3.16)$$

posteriormente descomponiendo los errores se logra:

$$\dot{\sigma}_{p_n} = k_{p_n} \dot{\xi}_{d_n} - k_{p_n} \dot{\xi}_n + \ddot{\xi}_{d_n} - \ddot{\xi}_n \quad (3.17)$$

y notando que en (3.17) aparece la ecuación del modelo dinámico (3.2), se procede a realizar la sustitución quedando como:

$$\dot{\sigma}_{p_n} = k_{p_n} \dot{\xi}_{d_n} - k_{p_n} \dot{\xi}_n + \ddot{\xi}_{d_n} - u_{p_n} - d_{\xi_n} \quad (3.18)$$

y dado que estos términos son dependientes del tiempo, es posible agruparlos tal que:

$$\rho_p(t, \sigma_{p_n}) = k_{p_n} \dot{\xi}_{d_n} - k_{p_n} \dot{\xi}_n + \ddot{\xi}_{d_n} - d_{\xi_n} \quad (3.19)$$

para que de esta manera la dinámica en términos de la variable deslizante descrita en (3.16) y el control ya definido en (3.15) se obtengan de la siguiente forma:

$$\begin{aligned} \dot{\sigma}_{p_n} &= \rho_p(t, \sigma_{p_n}) - C_{p_n} \sqrt{|\sigma_{p_n}|} \text{sgn}(\sigma_{p_n}) + w_{p_n} \\ \dot{w}_{p_n} &= B_{p_n} \text{sgn}(\sigma_{p_n}) \end{aligned} \quad (3.20)$$

3.1.4 Control para la dinámica de orientación de N agentes

Para el caso del control de orientación, el principal objetivo es buscar seguir una matriz de rotación deseada ya definida como R_{d_n} , la cual corresponde a los vectores f_{g_n} , b_{g_n} y n_{g_n} , además de la velocidad angular deseada la cual se define como:

$$\hat{\Omega}_{d_n} = R_{d_n}^\top \dot{R}_{d_n} \quad (3.21)$$

Debido a esto, es necesario definir los correspondientes errores de *posición angular* y *velocidad angular* tal que:

$$e_{R_n} = \text{Skew}(R_{d_n}^\top R_n)^\vee \quad (3.22)$$

$$e_{\Omega_n} = \Omega_n - \Omega_{d_n} \quad (3.23)$$

además, se define una distancia de medición entre las matrices de rotación *real* y *deseada*, esto es R_n y R_{d_n} como $\Upsilon(R_n R_{d_n})$. Esto significa que si ambas matrices son iguales, esta distancia es igual a cero, por lo que se cumpliría que:

$$\Upsilon(R_n R_{d_n}) = 0 \Leftrightarrow R_n = R_{d_n}$$

Establecido lo anterior, es posible definir dicha distancia o *función de error de rotación* entre las matrices R_n y R_{d_n} ; así como su razón de cambio como:

$$\Upsilon(R_n R_{d_n}^\top) = \frac{1}{2} \text{tr}(I - R_n R_{d_n}^\top) \quad (3.24)$$

$$\frac{d}{dt} \Upsilon(R_n R_{d_n}^\top) = \text{Skew}(R_{d_n}^\top R_n)^\vee (\Omega_n - \Omega_{d_n}) \quad (3.25)$$

donde se utilizan las siguientes operaciones:

- $\text{Skew}(A) = \frac{1}{2}(A - A^\top)$, esto es la obtención de una matriz *anti-simétrica* de A .
- $(.)^\vee$: corresponde al inverso del operador $(\hat{\cdot})$, lo que quiere decir que reacomoda los términos de la matriz A en un vector columna de tal manera que se cumpla que $a \times b = \hat{a}b$.

Posteriormente se procede a definir la superficie deslizante para el diseño del control utilizando los errores definidos en (3.22) y (3.23), por lo que la misma resulta como:

$$\sigma_{a_n} = k_{a_n} e_{R_n} + e_{\Omega_n} \quad (3.26)$$

Entonces, el control queda definido de la siguiente manera:

$$\begin{aligned} u_{a_n} &= C_{a_n} \sqrt{|\sigma_{a_n}|} \text{sgn}(\sigma_{a_n}) + w_{a_n} \\ \dot{w}_{a_n} &= B_{a_n} \text{sgn}(\sigma_{a_n}) \end{aligned} \quad (3.27)$$

donde C_{a_n} y B_{a_n} son matrices diagonales definidas positivas cuyos elementos son calculados como $c_{a_n,v} = 1.5\sqrt{H_{a_n}}$ y $b_{a_n,v} = 1.1H_{a_n}$ de donde H_{a_n} es una ganancia positiva, ver Apéndice A, y con $v = 1, 2, 3$ correspondiendo a las distintas acciones de control para cada eje de giro.

3.1.4.1 Dinámica resultante de orientación

La dinámica resultante del sistema se obtiene diferenciando la variable *deslizante* (3.26) de tal manera que se logra:

$$\dot{\sigma}_{a_n} = k_{a_n} \dot{e}_{R_n} + \dot{e}_{\Omega_n} \quad (3.28)$$

posteriormente descomponiendo los errores se tiene:

$$\dot{\sigma}_{a_n} = \frac{d}{dt} \text{Skew}(R_{d_n}^\top R_n)^\vee + \dot{\Omega}_n - \dot{\Omega}_{d_n} \quad (3.29)$$

y notando que en (3.29) aparece la ecuación del modelo dinámico (3.4), se procede a realizar la sustitución quedando como:

$$\dot{\sigma}_{a_n} = \frac{d}{dt} \text{Skew}(R_{d_n}^\top R_n)^\vee - \dot{\Omega}_{d_n} + u_{a_n} + d_{R_n} \quad (3.30)$$

y dado que estos términos son dependientes del tiempo, es posible agruparlos como:

$$\rho_R(t, \sigma_{a_n}) = \frac{d}{dt} \text{Skew}(R_{d_n}^\top R_n)^\vee - \dot{\Omega}_{d_n} + d_{R_n} \quad (3.31)$$

para que de esta manera la dinámica en términos de la variable deslizante descrita en (3.28) y el control ya definido en (3.27) se obtengan de la siguiente forma:

$$\begin{aligned} \dot{\sigma}_{a_n} &= \rho_R(t, \sigma_{p_n}) + C_{a_n} \sqrt{|\sigma_{a_n}|} \text{sgn}(\sigma_{a_n}) + w_{a_n} \\ \dot{w}_{a_n} &= B_{a_n} \text{sgn}(\sigma_{a_n}) \end{aligned} \quad (3.32)$$

Capítulo 4

Vuelo en Formación

En este capítulo se presentan los conceptos de vuelo en formación para cuadirrotores considerados agentes, así como los elementos necesarios para comprender el algoritmo de *Consensus*.

El vuelo en formación es una de las misiones que un conjunto de vehículos (multi-agentes)¹ realizan para poder llegar a un objetivo en común, que si se utilizara un solo agente, no sería posible realizar. Además, los trabajos y misiones se dan de manera más eficiente, y también es posible tener redundancia en el sistema para que de esta manera se garantice la seguridad de la misión y se tenga tolerancia a fallas.

Este tipo de vuelo es inspirado principalmente en el comportamiento colaborativo de los seres vivos, como manadas de animales, parvadas de pájaros, bancos de peces, colonias de bacterias, entre otras. En efecto, estas agrupaciones de animales tienen como fin común la realización de una tarea la cual tiene un beneficio conjunto para todos [1].

4.1 Reglas de Reynolds

El movimiento colectivo de grandes grupos de *agentes* debe de seguir o respetar ciertas reglas que gobiernan el comportamiento general de todo el sistema [21]. Es-

¹Término más comúnmente utilizado en el campo de la colaboración y cooperación de sistemas

to debido a que el movimiento dentro de un grupo obedece dos comportamientos principalmente, el hecho de mantenerse cerca de este grupo, y evitar colisiones con los demás individuos del mismo. Lo anterior fue establecido por *Reynolds* en [22] y puede ser descrito de manera general por las siguientes tres reglas que cada *agente* del grupo debe cumplir:

- **Evasión de colisiones:** es necesario que cada uno de los *agentes* del sistema evite colisiones con los demás, para esto se tienen que realizar los movimientos correctivos adecuados en caso del desvío de alguno de ellos, no solo por el *agente* afectado, si no por los agentes vecinos también.
- **Igualación de velocidad:** los miembros del grupo deben de igualar la velocidad y dirección del movimiento con respecto a sus vecinos para de esta manera respetar la regla anterior; es decir, evitar colisiones entre si.
- **Mantener distancia:** También es necesario mantener una distancia establecida entre los agentes, sin alejarse, para de esta manera alcanzar la *formación*, pero también sin acercarse demasiado para cumplir con la evasión de colisiones.

Debido a estas reglas, es necesario que en todo momento exista un flujo de información entre los agentes de manera que estos puedan conocer el estado y comportamiento de sus vecinos y así poder realizar las acciones correctivas necesarias para mantener la formación. Para describir este flujo de información, se utilizan lo que se conoce como *grafo*.

4.2 Teoría de Grafos

Un grafo es una representación tipo diagrama de una conexión o flujo de información entre individuos o *agentes*. Este está representado por una serie de *nodos* o *vértices*, que representan a los *agentes* del sistema, y sus correspondientes conexiones se establecen como *aristas* que van de un nodo a otro dependiendo de la conexión. En

la Figura 4.1 es posible observar un ejemplo, donde se tienen cuatro agentes que comparten información entre sí. Se puede notar que el nodo 1 comparte información con los nodos 2 y 3, así como ellos entre sí, pero el nodo 4 solo con el nodo 3.

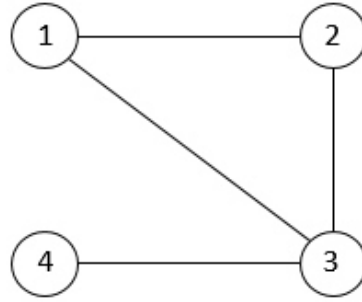


Figura 4.1: Grafo básico de conexión entre 4 agentes.

Entonces se puede notar que un *grafo de conexión* es quien representa aquellas conexiones que existen entre los nodos, y gracias al mismo es posible visualizar que nodos tienen flujo de información directamente entre ellos, y cuales no se conectan entre sí.

Por lo tanto, se puede definir a un *grafo* como un par $G = (V, E)$ donde V es una serie de N nodos definidos como $V = \{v_1, \dots, v_N\}$ y además E representa las aristas o líneas de conexión entre estos nodos y están denotados como $E = (v_i, v_j)$ que representa un flujo de datos proveniente desde i terminando en j , representado por flecha; esto es, con la punta de la misma en j y el inicio en i , ver Figura 4.2.

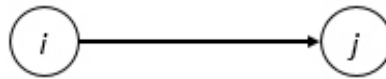


Figura 4.2: Representación del flujo de información de i hacia j , notese que la comunicación es unidireccional.

Además se establece que no existen aristas de conexión *reentrantes*, esto es $(v_i, v_i) \notin E, \forall i$, y que las conexiones existentes no son múltiples; es decir, no se repiten varias veces. Por consiguiente, se definen algunos conceptos o características básicas de los grafos que ayudan a la identificación de los mismos:

- **Entrante y Saliente:** es referido a la naturaleza de la arista con respecto al nodo visto; esto es, una arista (v_i, v_j) es *Saliente* del nodo i y *Entrante* al nodo j .
- **Grado de Entrada de un nodo:** es la cantidad de aristas que tienen como punta al nodo en cuestión, en otras palabras, es el número de conexiones entrantes a un nodo.
- **Grado de Salida de un nodo:** es la cantidad de aristas que tienen como inicio al nodo en cuestión; en otras palabras, es el número de conexiones salientes desde un nodo.
- **Bi-direccionalidad:** propiedad de las aristas que tienen comunicación entrante y saliente para ambos nodos que conectan, esto se expresa como: $(v_i, v_j) \in E \Rightarrow (v_j, v_i) \in E, \forall i, j$
- **Peso de la arista:** Definido como a_{ij} , es el grado de conexión que se tiene en la comunicación del nodo j hacia el i ; esto es, si $a_{ij} = 1$ significa que existe la comunicación, en caso de ser 0, no existe comunicación. Además se asumen valores estrictamente positivos, y en caso de ser diferente de 1, el peso se toma como factor multiplicativo en la transmisión de información. Por lo tanto, la comunicación se puede clasificar como:
 - *No dirigida:* Se da si $a_{ij} = a_{ji}$, esto es, que la comunicación de i con j , existe de manera entrante y saliente para ambas direcciones.
 - *Dirigida:* Ocurre cuando $a_{ij} \neq a_{ji}$, por lo que la comunicación tiene prioridad en alguno de los sentidos, o en caso de que alguno de los pesos sea 0, es inexistente para alguno de ellos.

4.3 Matrices de Grafos

Para poder comprender mejor la información que un grafo provee, es posible definir ciertas propiedades haciendo uso de matrices, las cuales contienen esta información de manera más ordenada y esquemática. Estas matrices se presentan a continuación.

4.3.1 Matriz de Adyacencia o Conectividad

Como su nombre lo indica, esta matriz representa las conexiones que se tienen entre los nodos que forman el sistema. Esta es representada por los pesos establecidos como $A = [a_{ij}]$ con dichos pesos como $a_{ij} > 0$ si $(v_j, v_i) \in E$. El *grado de entrada* de un nodo v_i correspondería a la suma de los pesos a_{ij} de la i – *esima* fila de la matriz de adyacencia A , mientras que el *grado de salida* sería a su vez la suma de los pesos a_{ji} de la i – *esima* columna de la misma matriz.

En otras palabras, un vector fila de la matriz A contiene los pesos de las conexiones de entrada correspondientes al nodo descrito por la fila, tomando el número de columna como el nodo con quien se tiene la conexión, y un vector columna contiene los pesos de salida del nodo tomando el número de fila como el nodo a quien se le envía información.

Para la Figura 4.3 la matriz de Adyacencia A esta dada como:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

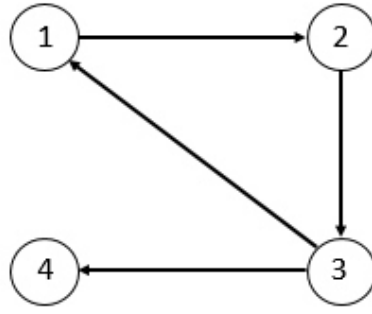


Figura 4.3: Representación de un Grafo Dirigido.

4.3.2 Matriz Laplaciana

La matriz Laplaciana es de gran importancia dado que se la misma se pueden estudiar varias características importantes de un grafo. Está definida como $L = D - A$, donde A es la *matriz de Adyacencia* y D es una matriz diagonal formada con los *grados de entrada* de los nodos correspondientes. Tiene como propiedad que la suma de todas sus filas siempre es igual a cero. Tomando como ejemplo el grafo dirigido de la Figura 4.3, la matriz D , y por consiguiente la matriz Laplaciana L corresponden a:

$$D = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad L = \begin{bmatrix} 1 & 0 & -1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

4.4 Algoritmo de *Consensus*

Para poder realizar la coordinación de los distintos vehículos pertenecientes al sistema de *multi-agentes*, es necesario traducir las *Reglas de Reynolds* a una forma

utilizable matemáticamente o como un algoritmo. Además, se tiene que considerar las conexiones que se tienen entre los agentes, y por ello definir herramientas y parámetros como es el caso de la *matriz de adyacencia*. De igual manera, para poder definir una expresión matemática que describa el comportamiento deseado, es necesario considerar la dinámica de los *agentes* para tener el control correcto de sus parámetros.

Por consiguiente, el algoritmo tiene que hacer que los *agentes* del sistema lleguen a una coordinación o «acuerdo» con respecto a sus estados, y en base a la información intercambiada, decidir la acción a realizar. Lo anterior es llamado *Consensus*² y describe el comportamiento coordinado y toma de decisiones en los *agentes* de un grupo.

4.4.1 *Consensus* en dinámicas de segundo orden

Suponiendo que todos los *agentes* del sistema tienen el comportamiento dinámico correspondiente a un *doble integrador*; es decir, de segundo orden similar a que puede ser visto en (A.1), el cual corresponde al siguiente conjunto de ecuaciones:

$$\begin{aligned}\dot{x}_{i1} &= x_{i2} \\ \dot{x}_{i2} &= u_i\end{aligned}$$

es posible definir un protocolo de control para cada uno de los agentes, con dinámica similar, como:

$$u_i = k_p \sum_{j \in N_i} a_{ij}(x_{j1} - x_{i1}) + k_v \sum_{j \in N_i} a_{ij}(\dot{x}_{j1} - \dot{x}_{i1}) \quad (4.1)$$

donde k_p es la ganancia de la parte del protocolo encargada de mantener o igualar las posiciones de los *agentes*, y k_v es la ganancia de la parte correspondiente a la coordinación de velocidad. Además, a_{ij} es el ij – *esimo* elemento de la matriz de

²Del término en inglés para la palabra *Consensus* que significa llegar a un acuerdo.

adyacencia A , lo cual representa si existe comunicación proveniente de tal elemento j . Finalmente N_i es el conjunto de agentes vecinos a i ; es decir, el resto de agentes del sistema.

El protocolo (4.1), al ser utilizado como única ley de control, fuerza a los *agentes* a adquirir una misma posición y velocidad a medida que avanza el tiempo. Esto debido a que u_i sería cero solo cuando los estados x_{i_1} y \dot{x}_{i_1} son iguales a sus contrapartes del vecino j , esto es, a x_{j_1} y \dot{x}_{j_1} .

Sabiendo que $\dot{x}_1 = x_2$, el sistema en lazo cerrado puede desarrollarse como:

$$\dot{x}_{i_2} = -k_p x_{i_1} \sum_{j \in N_i} a_{ij} + k_p \sum_{j \in N_i} a_{ij} x_{j_1} - k_v x_{i_2} \sum_{j \in N_i} a_{ij} + k_v \sum_{j \in N_i} a_{ij} x_{j_2} \quad (4.2)$$

Además, de (4.2) se puede notar que $\sum a_{ij}$ para todo $j \in N_i$ es el grado de entrada d_i para el nodo i , por lo que se puede reescribir como:

$$\dot{x}_{i_2} = -k_p x_{i_1} d_i + k_p \sum_{j \in N_i} a_{ij} x_{j_1} - k_v x_{i_2} d_i + k_v \sum_{j \in N_i} a_{ij} x_{j_2} \quad (4.3)$$

Finalmente, de (4.3) se expresan las sumatorias como un producto de vectores quedando como:

$$\dot{x}_{i_2} = -k_p x_{i_1} d_i + k_p [a_{i_1} \cdots a_{N_1}] \begin{bmatrix} x_{1_1} \\ \vdots \\ x_{N_1} \end{bmatrix} - k_v x_{i_2} d_i + k_v [a_{i_2} \cdots a_{N_2}] \begin{bmatrix} x_{1_2} \\ \vdots \\ x_{N_2} \end{bmatrix} \quad (4.4)$$

De esta manera, (4.4) resulta en la dinámica de lazo cerrado para un solo nodo del sistema. Para el caso del sistema global, se define un vector de estados $x_G = [x_1 \cdots x_N]^T$ y se definen las siguientes partes de la ecuación (4.4):

$$\dot{x}_{i_2} = -k_p x_{i_1} d_i + k_p \underbrace{[a_{i_1} \cdots a_{N_1}]}_a \underbrace{\begin{bmatrix} x_{1_1} \\ \vdots \\ x_{N_1} \end{bmatrix}}_{b_1} - k_v x_{i_2} d_i + k_v \underbrace{[a_{i_2} \cdots a_{N_2}]}_a \underbrace{\begin{bmatrix} x_{1_2} \\ \vdots \\ x_{N_2} \end{bmatrix}}_{b_2}$$

Es posible resaltar que las partes señaladas como a corresponden a la fila i de la matriz de adyacencia A , por lo que se puede tomar como una ecuación global para N nodos. Además b_1 corresponde al vector de estados ya definido como x_G , y b_2 corresponde a su primera derivada; esto es, \dot{x}_G . Entonces, sabiendo que la matriz diagonal de grados de entrada está dada por $D = \text{diag}\{d_i\}$ y que la matriz *Laplaciana* es $L = D - A$, se puede establecer la ecuación en lazo cerrado del sistema global como:

$$\begin{aligned} \ddot{x}_G &= -k_p D x_G + k_p A x_G - k_v D \dot{x}_G + k_v A \dot{x}_G \\ \ddot{x}_G &= k_p (-D x_G + A x_G) + k_v (-D \dot{x}_G + A \dot{x}_G) \\ \ddot{x}_G &= k_p [(-D + A) x_G] + k_v [(-D + A) \dot{x}_G] \\ \ddot{x}_G &= k_p (-L x_G) + k_v (-L \dot{x}_G) \\ \ddot{x}_G &= -L(k_p x_G + k_v \dot{x}_G) \end{aligned} \tag{4.5}$$

4.4.2 Algoritmo *Consensus no-lineal* por medio de control *Super Twisting*

Para poder alcanzar el «consensus» es necesario que los estados de los agentes en cuestión lleguen a un mismo valor, o en su debido caso, a la misma condición previamente establecida, como por ejemplo a la igualación de los errores de seguimiento. Este objetivo se puede cumplir con el protocolo de control propuesto en (4.1), y además, es posible utilizarlo como ley de control única en el sistema de lazo cerrado y de esta manera garantizar el cumplimiento del «consensus». Sin embargo,

este protocolo no garantiza una respuesta rápida a cambios bruscos en la referencia de seguimiento, así como un buen rechazo a las perturbaciones, o dependiendo del caso, una rápida convergencia a la respuesta sin presencia de sobre-tiro indeseado. Debido a las desventajas que conlleva la utilización del protocolo (4.1) de manera única³, en este trabajo se propone una aproximación de este protocolo por medio de la utilización del control por *modos deslizantes* utilizando el algoritmo de *Super Twisting*.

Considerando el protocolo (4.1) establecido como:

$$u_i = k_p \sum_{j \in N_i} a_{ij} (x_{j1} - x_{i1}) + k_v \sum_{j \in N_i} a_{ij} (\dot{x}_{j1} - \dot{x}_{i1})$$

es posible reescribirlo de tal manera que los agentes de la formación, en vez de igualar sus estados, busquen tener el mismo error de seguimiento tanto en posición como en velocidad. Lo anterior permite que los agentes mantengan una formación, la cual es alimentada de manera independiente.

Definiendo los errores de seguimiento para la parte de la coordinación de la siguiente manera:

$$\begin{aligned} e_{c_j} &= x_j - x_{ref_j} \\ e_{c_i} &= x_i - x_{ref_i} \end{aligned} \tag{4.6}$$

$$\begin{aligned} \dot{e}_{c_j} &= \dot{x}_j - \dot{x}_{ref_j} \\ \dot{e}_{c_i} &= \dot{x}_i - \dot{x}_{ref_i} \end{aligned} \tag{4.7}$$

donde el conjunto de ecuaciones (4.6) corresponde a los errores de seguimiento de posición donde x_{ref} corresponde al *estado de posición deseado* o de *referencia*, mientras que las ecuaciones (4.7) representan el error de seguimiento de velocidad. Por lo

³Es posible notar que el protocolo (4.1) tiene la forma básica de un control *PD* (*Proportional-Derivativo*); esto es, se tiene un término proporcional del error (P), así como un término de velocidad (D), y se utilizan las ganancias correspondientes para llevar los estados a la referencia. Bajo ciertas condiciones este tipo de aproximación es suficiente; sin embargo se comienzan a presentar desventajas, principalmente cuando el sistema tiende a ser completamente no-lineal.

tanto \dot{x}_{ref} es el *estado de velocidad deseada* o de *referencia*, y es posible de reescribir el protocolo de formación como:

$$u_i = k_p \sum_{j \in N_i} a_{ij}(e_{c_j} - e_{c_i}) + k_v \sum_{j \in N_i} a_{ij}(\dot{e}_{c_j} - \dot{e}_{c_i}) \quad (4.8)$$

donde ahora e_{c_j} y e_{c_i} representan el error de seguimiento de posición de los agentes j e i correspondientemente; mientras que sus derivadas son los errores de velocidad. Por lo tanto, el protocolo (4.8) garantiza que si uno de los agentes se desvía de su trayectoria original, o es perturbado de alguna manera, los demás van a seguirlo tratando de mantener la formación previamente definida.

Teniendo el protocolo (4.8), se procede a reescribirlo de manera que se puedan agrupar de manera más sencilla los errores de seguimiento para cada nodo, resultando en la siguiente ecuación.

$$u_i = -k_p e_{c_i} \sum_{j \in N_i} a_{ij} + k_p \sum_{j \in N_i} a_{ij} e_{c_j} - k_v \dot{e}_{c_i} \sum_{j \in N_i} a_{ij} + k_v \sum_{j \in N_i} a_{ij} \dot{e}_{c_j} \quad (4.9)$$

Debido a la forma general que adopta la ecuación (4.9), se propone el diseño de una *superficie deslizante* utilizándola, para que de esta manera sea posible la coordinación del sistema de *multi-agentes* por medio de una ley de control *Super Twisting*. Además del arreglo descrito para el control de un solo agente, se añade a la superficie deslizante los términos de la «coordinación» respetando la estructura del error de seguimiento (A.2) así como la superficie deslizante clásica en (A.3), por lo que se procede a realizar el acomodo de términos semejantes para tener la estructura afín.

Primero, se define un *término de coordinación* σ_c con los elementos de (4.9), el cual será parte de la superficie deslizante general de control. Realizando una agrupación tomando en cuenta los términos correspondientes a cada nodo se tiene:

$$\sigma_c = -k_p e_{c_i} \sum_{j \in N_i} a_{ij} - k_v \dot{e}_{c_i} \sum_{j \in N_i} a_{ij} + k_p \sum_{j \in N_i} a_{ij} e_{c_j} + k_v \sum_{j \in N_i} a_{ij} \dot{e}_{c_j}$$

$$\sigma_c = - \sum_{j \in N_i} a_{ij} (k_p e_{c_i} + k_v \dot{e}_{c_i}) + \sum_{j \in N_i} a_{ij} \left(k_p \sum_{j \in N_i} e_{c_j} + k_v \sum_{j \in N_i} \dot{e}_{c_j} \right) \quad (4.10)$$

Ahora, con la forma el error mostrada en (A.2), es posible notar que contrasta *signos* con respecto a los errores de coordinación en (4.6) y (4.7). Dado que se necesita tener la estructura del primero para poder continuar con la convención de signos (principalmente en las ganancias del controlador), los signos de la ecuación (4.10) se modifican para que se obtenga la equivalencia deseada.

$$\sigma_c = \sum_{j \in N_i} a_{ij} (k_p e_{c_i} + k_v \dot{e}_{c_i}) - \sum_{j \in N_i} a_{ij} \left(k_p \sum_{j \in N_i} e_{c_j} + k_v \sum_{j \in N_i} \dot{e}_{c_j} \right) \quad (4.11)$$

Finalmente, se establece la *superficie deslizante general* σ_g la cual es el resultado de la suma de la superficie deslizante, para el control de navegación del agente individual; esto es, σ , definida en (A.3), y el término de compensación σ_c .

$$\sigma_g = \sigma + \sigma_c \quad (4.12)$$

Entonces, el procedimiento de control utilizado es similar al que se describe en (A.7) quedando la ley de control como:

$$u_i = c_n \sqrt{|\sigma_g|} \operatorname{sgn}(\sigma_g) + w$$

$$\dot{w} = b[\operatorname{sgn}(\sigma_g)] \quad (4.13)$$

utilizando las siguientes relaciones para calcular las ganancias requeridas:

$$b = 1.1 H_n$$

$$c_n = 1.5 \sqrt{H_n}$$

Capítulo 5

Plataforma experimental

En este capítulo se presenta la plataforma experimental utilizada para la validación e implementación en tiempo real de los algoritmos desarrollados. Para realizar esto se utilizaron principalmente tres componentes, el VANT que serviría como agente, un sistema para la obtención de sus estados, y una interfaz de control. Para lo anterior se utilizó un cuadrirrotor fabricado por la compañía francesa *Parrot*® y específicamente el modelo *AR. Drone 2.0*®. De igual manera, para la obtención de las variables de estado, fue utilizado un sistema de captura de movimiento *VICON*®. Además como interfaz entre estos elementos, se diseñó una *estación tierra* utilizando la plataforma de desarrollo *LabVIEW*® de la compañía *National Instruments*®. Estos componentes, junto con los distintos elementos necesarios para completar la implementación, la arquitectura y arreglo de los mismos en el entorno de trabajo son descritos a continuación.

5.1 Cuadrirrotor *Parrot*® *AR. Drone 2.0*®

El *AR. Drone 2.0*® es un quadri-rotor comercial cuya arquitectura física corresponde a la de un cuadrirrotor en configuración tipo **X**. El uso principal de este vehículo es meramente recreativo; sin embargo, dada su construcción resistente y duradera, así como como las distintas ventajas en su maniobrabilidad, lo hacen una excelente herramienta para fines de investigación. Entre algunas de las características desta-

características del vehículo se tienen [23]:

- Comunicación a través de protocolos *Wi-Fi* 802.11b/g/n con hasta 50 metros de alcance.
- Transmisión de video en vivo con 720p de resolución a 30 imágenes por segundo.
- Sensor de altitud compuesto por un sensor ultrasónico y de presión del aire.
- Unidad de medición inercial compuesta de giroscopio, acelerómetro y magnetómetro de 3 ejes cada uno.
- Construcción ligera en nylon y fibra de carbono, así como cubiertas de poliestireno.
- Dos cubiertas intercambiables, una para interiores y otra para exteriores que reduce efectos aerodinámicos sobre el cuadrirrotor.

En la Figura 5.1 se muestra uno de los *AR. Drone 2.0*® utilizados durante este trabajo, utilizando la cubierta para interiores.



Figura 5.1: Cuadrirrotor utilizado para la realización de experimentos *AR. Drone 2.0*®

5.2 Sistema de captura de movimiento *VICON*®

Para realizar la lectura de las variables de estado de los vehículos se utiliza un sistema de captura de movimiento fabricado por la empresa *VICON*®. Este sistema consiste en 16 cámaras *T-40* las cuales se encargan de capturar los datos de movimiento y enviarlos a una computadora central por medio de *conmutadores Gigabit Ethernet (MX GIAGANET)*.

5.2.1 Hardware, Cámaras *VICON*® *T-40* y *GIGANET*

El sistema utilizado para la realización de este trabajo consiste en 16 cámaras *T-40*, ver Figura 5.2. Estas cámaras pueden capturar distintos tipos de marcadores reflejantes, ver Figura 5.3, a velocidades de hasta 370 imágenes por segundo.



Figura 5.2: Cámara *VICON*® *T-40*



Figura 5.3: Elementos reflejantes que las cámaras del sistema siguen

Dadas las capacidades de estas cámaras (ver Tabla 5.1 para más especificaciones) es posible realizar una adquisición de los estados de posición lineal y angular con la precisión para poder estimar velocidad y aceleración con adecuada exactitud y utilizar estos datos en la implementación de los algoritmos propuestos.

Especificación	Dato
Máxima velocidad de captura a máxima resolución	370 fps
Resolución del sensor óptico	2352 x 1728 (4MP)
Tipo de sensor	CMOS
Exactitud en movimientos traslacionales	hasta 0.1 mm
Exactitud en movimientos rotacionales	hasta 0.1 grados

Tabla 5.1: Especificaciones técnicas de las cámaras *T-40*.

Para enviar estos datos a una computadora central que haga el post-procesamiento de la información se necesita un conmutador *Gigabit ethernet* o *MX GIGANET*, ver Figura 5.4. Este conmutador distribuye los datos de hasta un máximo de 10 cámaras y además, si se requiere un número mayor, es posible interconectar varios con arreglos «maestro-esclavo» y de esta manera distribuir los datos de todos los conmutadores a la computadora central.



Figura 5.4: Conmutador ethernet modelo *MX GIGANET*.

5.2.2 Software de adquisición de datos *VICON Tracker*®

El software *VICON Tracker*® procesa la información capturada a través de los conmutadores *GIGANET*, ver Figura 5.5.



Figura 5.5: Software de adquisición de datos *VICON Tracker*®.

Este software permite tener el control de todas las cámaras del sistema así como de los parámetros de operación que es posible modificar, tales como:

- Intensidad de los estrobos
- Modo de escala de grises
- Frecuencia de refresco de imagen
- Cantidad de cuadros enviados al *buffer* de comunicación

De igual manera, posee una interfaz gráfica en la cual es posible visualizar en tiempo real los datos capturados por el sistema, desde una visualización 3D del entorno capturado y los objetos seguidos, Figura 5.6, hasta los datos individuales de cada cámara o de los marcadores que estén en el entorno de visibilidad para el sistema.

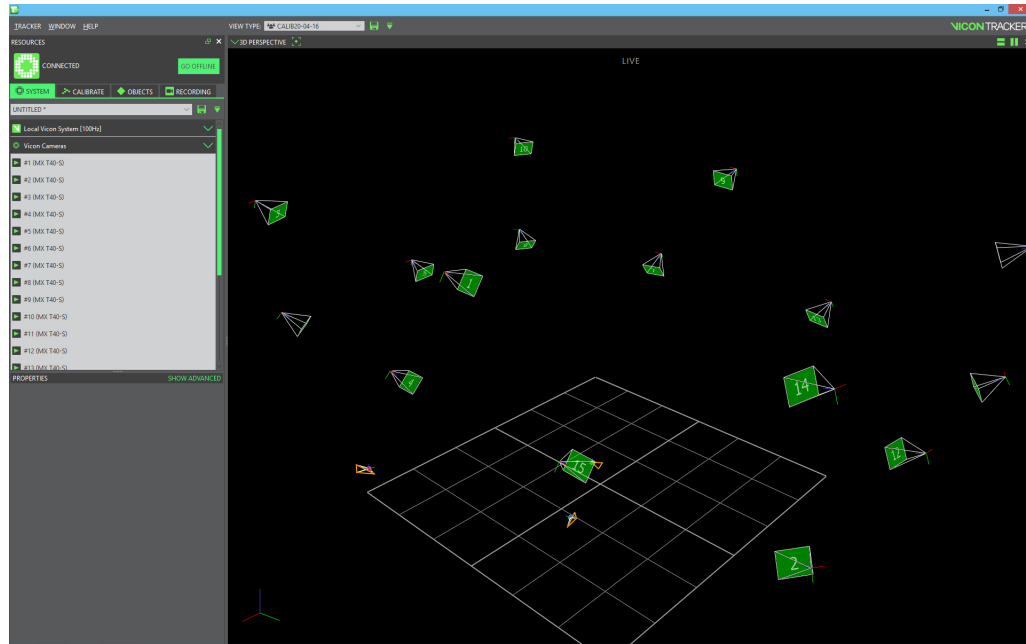


Figura 5.6: Visualización 3D del entorno capturado en el software *VICON Tracker*[®].

5.3 Plataforma de desarrollo *LabVIEW*[®]

Para realizar la integración del agente cuadrirrotor y del sistema de captura de movimiento, se diseñó una interfaz de comunicación haciendo uso de la plataforma de desarrollo *LabVIEW*[®]. Entre algunas de las características de este software se tienen:

- Programación de sistemas en entorno visual gráfico (utilizando bloques).
- Capacidad de conexión con otras plataformas comerciales como *MATLAB*[®].
- Variedad de interfaces de comunicación.
- Capacidad de interacción con otros lenguajes de programación. Además es posible escribir partes de código, como lenguaje *C*, dentro del mismo entorno de trabajo.

- Herramientas para interacción con tarjetas de adquisición de datos y demás componentes de hardware externos.

Sobre esta plataforma se realiza el *Guiado, Navegación y Control* de las distintas aeronaves utilizadas en las pruebas experimentales. Además, dadas las capacidades del software, se diseñan las distintas interfaces gráficas para poder comunicar de manera más intuitiva los datos de vuelo al operador, así como para enviar de manera más fácil las ordenes del mismo a los VANTs. Un ejemplo de una de las interfaces utilizadas puede ser visto en la Figura 5.7.

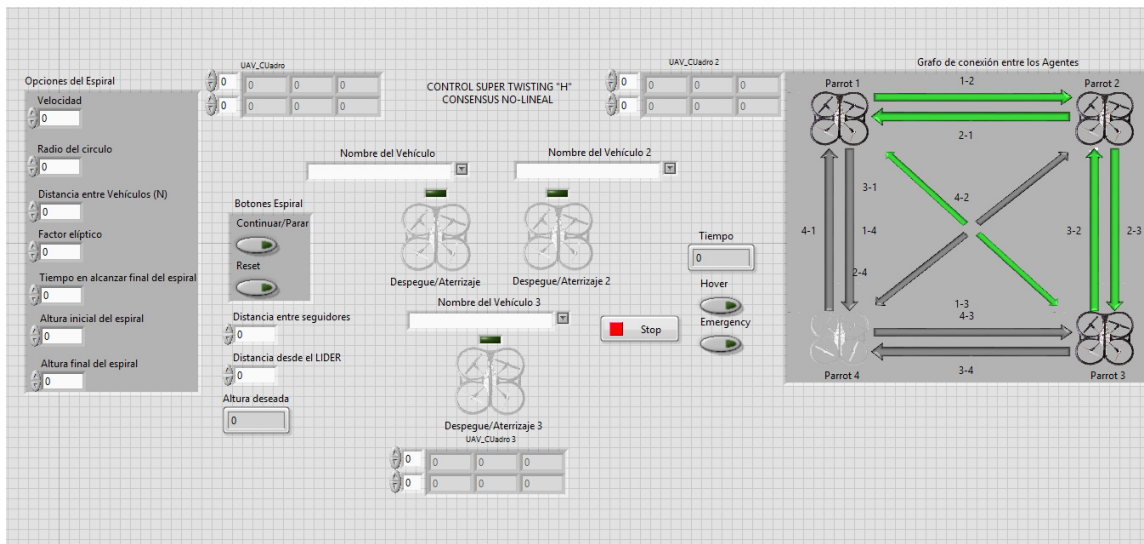


Figura 5.7: Ejemplo de interfaz utilizada en el software de desarrollo.

5.4 Arquitectura del entorno de trabajo

Para poder realizar la adquisición de datos y luego el procesamiento de los mismos y de esta manera obtener el resultado deseado en los cuadricópteros, es necesario tener un arreglo de los distintos componentes de hardware y software que componen el entorno de trabajo de tal manera que estos elementos se comuniquen adecuadamente y la información fluya de manera correcta.

Utilizando la información ya expuesta, es posible enlistar aquellos dispositivos que componen el entorno de trabajo y agregar aquellos accesorios necesarios para llegar al resultado deseado. Estos dispositivos son:

- Cámaras *VICON*® *T-40*.
- Conmutadores ethernet *MX GIGANET*.
- Computadora central de adquisición de datos.
- Conmutador ethernet (*switch*).
- Computadora de procesamiento y GNC.
- Enrutador *Wi-Fi*.
- AR. Drone 2.0®

Los dispositivos anteriores están en orden de ejecución, esto quiere decir que algorítmicamente el proceso se realiza como:

1. Se capturan los datos de traslación y orientación de los marcadores reflejantes que estén en el área de trabajo.
2. La información de las cámaras es enviada al conmutador *MX GIGANET* que concatena en una sola cadena de datos la información y la envía a la computadora central.
3. La computadora central realiza el procesamiento de los datos de las cámaras por medio del software *Tracker*® y los traduce a coordenadas de traslación y orientación en el centro de gravedad del agente o vehículo dependiendo de los marcadores capturados. Posteriormente esta información se organiza en arreglos de datos y se envía a la computadora cliente por medio del *switch*.
4. El *switch* distribuye los datos a las distintas computadoras clientes conectadas.

5. La computadora cliente, utilizando los datos recibidos, realiza el procesamiento de *Guiado, Navegación y Control* y envía las señales de control necesarias a los agentes cuadrirrotos correspondientes a través del *enrutador*.
6. El *enrutador* recibe los datos de control, y de acuerdo a las direcciones *IP* que contengan, decide a que cuadrirrotor enviar dicha información.
7. El cuadrirrotor recibe las señales de control y realiza las acciones y correcciones necesarias de acuerdo a dicha información.

Posteriormente, el ciclo se reinicia.

Lo anterior se puede apreciar esquemáticamente en la Figura 5.8.

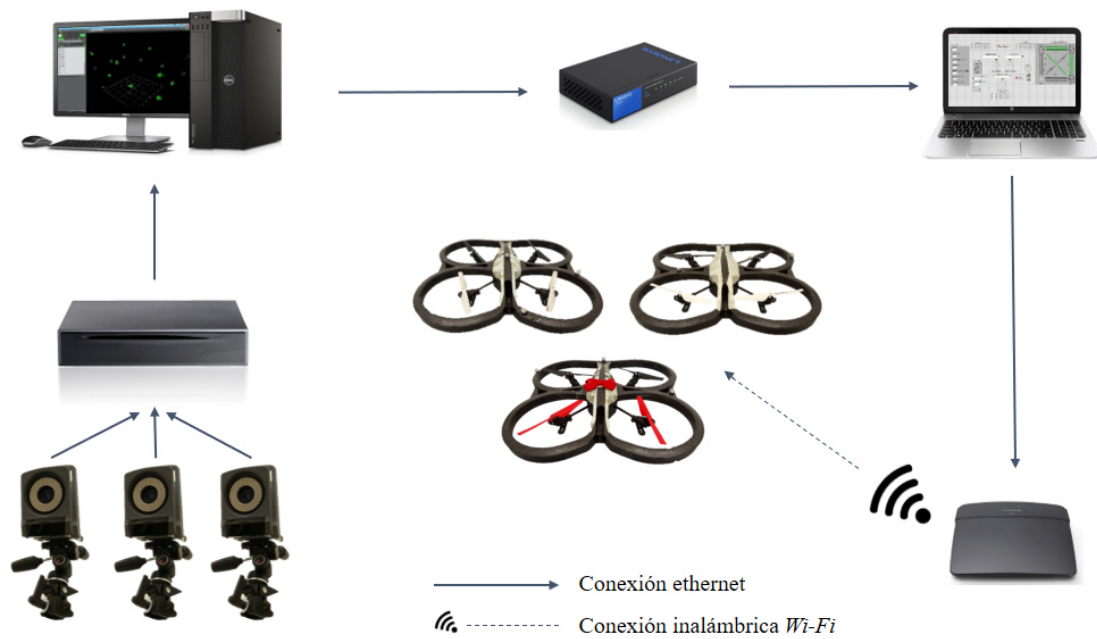


Figura 5.8: Arquitectura esquemática del entorno de trabajo.

Capítulo 6

Implementación en tiempo real

En este capítulo se abordan los experimentos realizados para validar las ecuaciones y algoritmos propuestos. Se realizan una serie de pruebas con el control *Super Twisting* para evaluar su desempeño. Posteriormente se procede a implementar un algoritmo de *Consensus* similar al descrito en (4.5) y finalmente se realiza la validación del algoritmo propuesto en la sección 4.4.2. Los experimentos realizados se enlistan a continuación:

- Pruebas del control *Super Twisting*.
 - Trayectoria circular.
 - Trayectoria en espiral.
 - Respuesta a perturbaciones físicas.
- *Consensus* de segundo orden.
 - Trayectoria circular con dos agentes (con perturbación).
 - Trayectoria para cubrir un área cuadrada.
- Propuesta de *Consensus no-lineal* en ley de control *Super Twisting*.
 - Trayectoria circular con dos agentes (con perturbación).
 - Trayectoria para cubrir un área cuadrada.

- Trayectoria en forma de *lemniscata*¹ con tres agentes en persecución.

Estos experimentos son realizados tomando en cuenta el ángulo de guiñada o de dirección del cuadrirrotor como se observa en la Figura 6.1, debido al sistema de coordenadas utilizado en el laboratorio donde realizó la implementación.

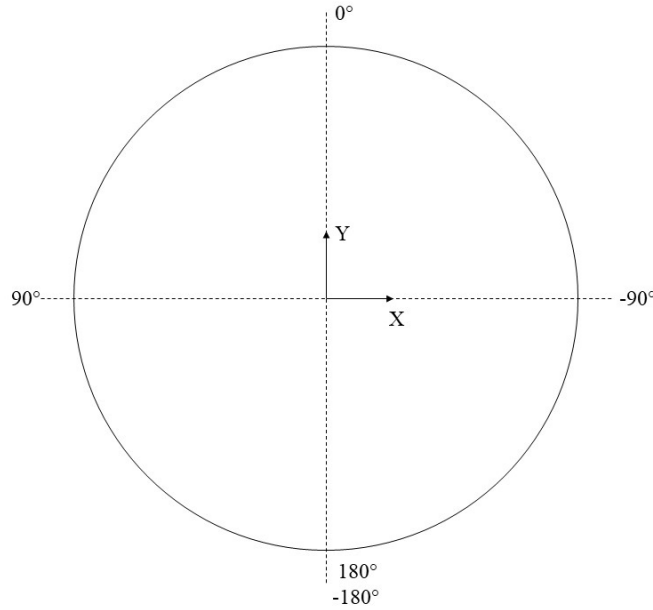


Figura 6.1: Ángulos de guiñada con respecto al sistema de coordenadas del laboratorio

6.1 Pruebas del control *Super Twisting*

Para la realización de las pruebas del controlador *Super Twisting* se procede a realizar la implementación de la ecuación vectorial (3.15) para el caso del control de posición, y a su vez la ecuación (3.27), correspondiente al control de orientación. Haciendo uso de los componentes mencionados en el Capítulo 5, se procede a la implementación de las ecuaciones de control, y a la realización de una serie de pruebas para evaluar el desempeño del cuadrirrotor en el seguimiento de trayectorias.

¹Simbolo de *infinito*

A continuación se presentan una serie de experimentos correspondiente a diferentes trayectorias las cuales son implementadas en tiempo real.

6.1.1 Trayectoria circular

Para poder visualizar el desempeño del control *Super Twisting* se asigna una trayectoria circular a un cuadrirrotor. Esto permite que las acciones de control U_x y U_y actuen de manera mezclada e intercambiándose entre sí dependiendo del ángulo de *guiñada* del vehículo.

La trayectoria es generada utilizando las ecuaciones paramétricas descritas como:

$$\begin{aligned}x &= x_{centroide} + r\sin(2\pi ft) \\ y &= y_{centroide} + r\cos(2\pi ft)\end{aligned}\tag{6.1}$$

donde x y y representan a las coordenadas en el marco inercial fijo de la trayectoria. Además, $x_{centroide}$ y $y_{centroide}$ corresponden a las coordenadas donde está el centro del círculo descrito, mientras que f , t y r corresponden a las *revoluciones por segundo*, al tiempo y al radio de la circunferencia, respectivamente. Para el caso de este experimento, los siguientes datos son utilizados:

- Radio del círculo de 1500 *mm*.
- Coordenadas del centro del círculo en:
 - $x_{centroide} = 0$ *mm*.
 - $y_{centroide} = 1000$ *mm*.
- Velocidad de 0.04 *rev/seg* (2.4 *rpm*).

Tomando los resultados de las ecuaciones (6.1) como entrada de posición para el control del cuadrirrotor, se obtiene como resultado el seguimiento del círculo deseado.

Los resultados son mostrados en la Figura 6.2, donde además se pueden observar el despegue y aterrizaje del cuadrirrotor.

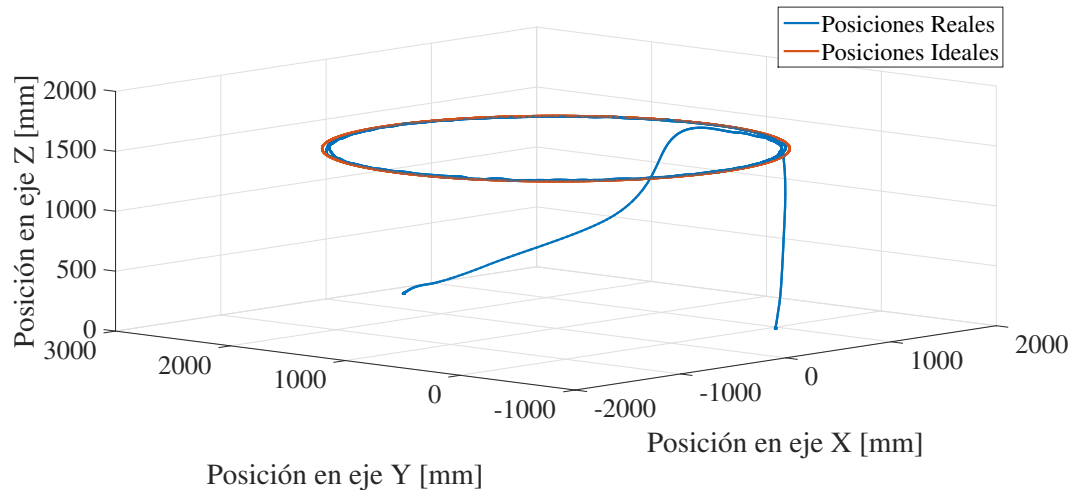


Figura 6.2: Trayectoria circular en el espacio tridimensional.

En la Figura 6.3 es posible observar el ángulo de *guiñada* que sigue el cuadrirrotor debido al control de orientación propuesto en (3.27). Además, es necesario resaltar que este ángulo representa un vector *tangente* a la trayectoria circular dado el marco de guiado en el espacio euclidiano $SE(3)$, respetando los ángulos de orientación del laboratorio utilizado (ver Figura 6.1).

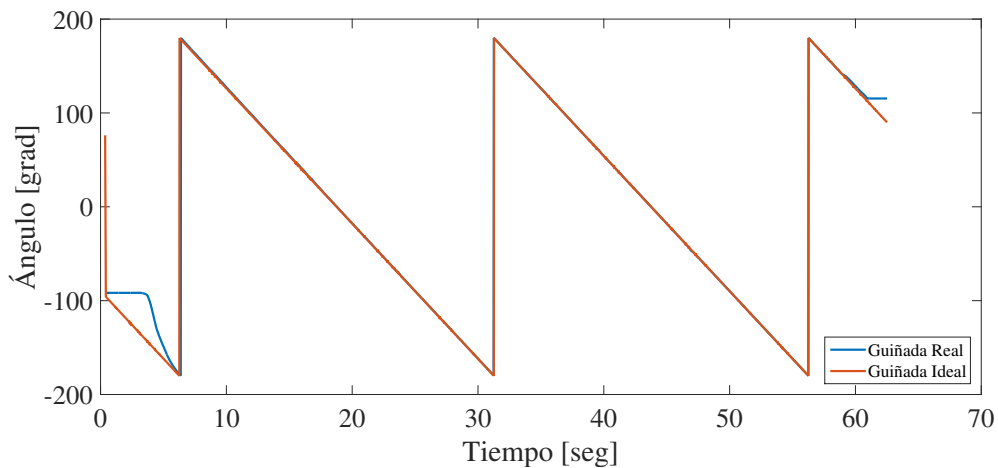
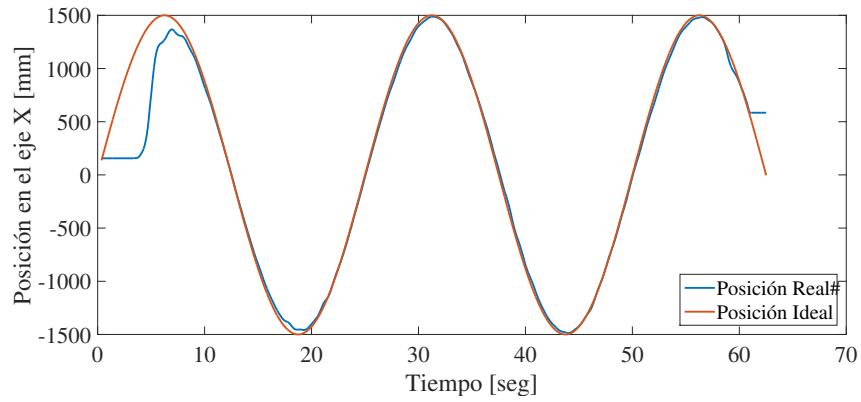
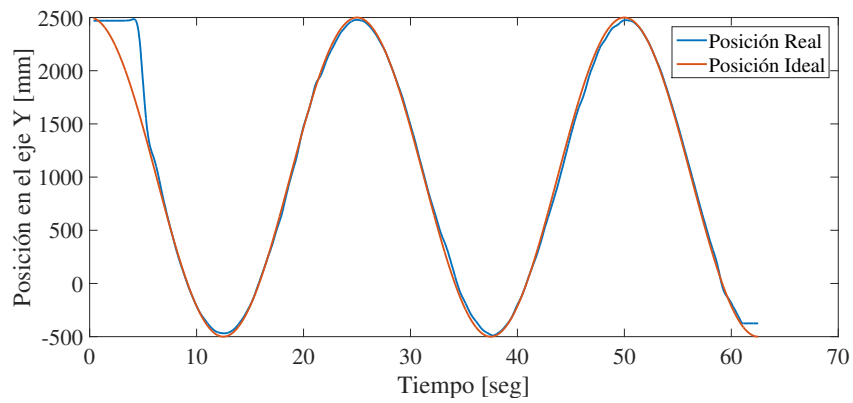


Figura 6.3: Ángulo de guiñada del cuadrirrotor siguiendo el círculo.

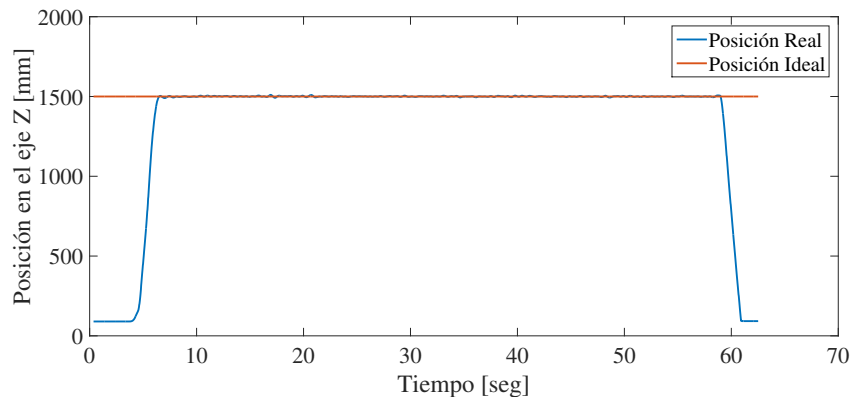
Para poder visualizar de mejor manera el comportamiento del cuadrirrotor con respecto al seguimiento de la trayectoria deseada, en la Figura 6.4 se muestran individualmente los resultados del seguimiento para cada eje de movimiento.



(a) Posición en el eje X.



(b) Posición en el eje Y.



(c) Posición en el eje Z.

Figura 6.4: Gráficas de posiciones XYZ por eje para la trayectoria circular.

Las *Superficies deslizantes* generadas son mostradas en la Figura 6.5, las cuales se calcularon a partir de las ecuaciones (3.14) y (3.26).

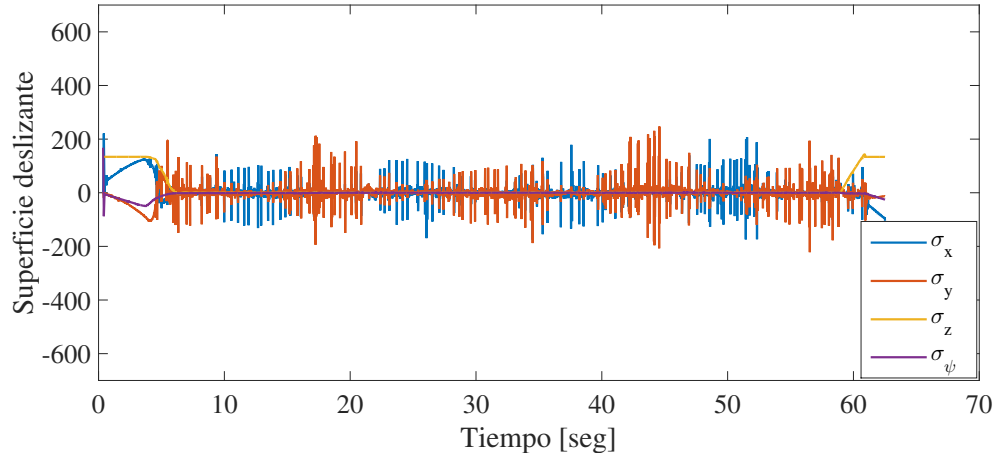


Figura 6.5: Superficies deslizantes generadas utilizadas para el control en la trayectoria circular.

Utilizando estas superficies deslizantes en el algoritmo *Super Twisting*, se generan las señales de control las cuales son mostradas en la Figura 6.6.

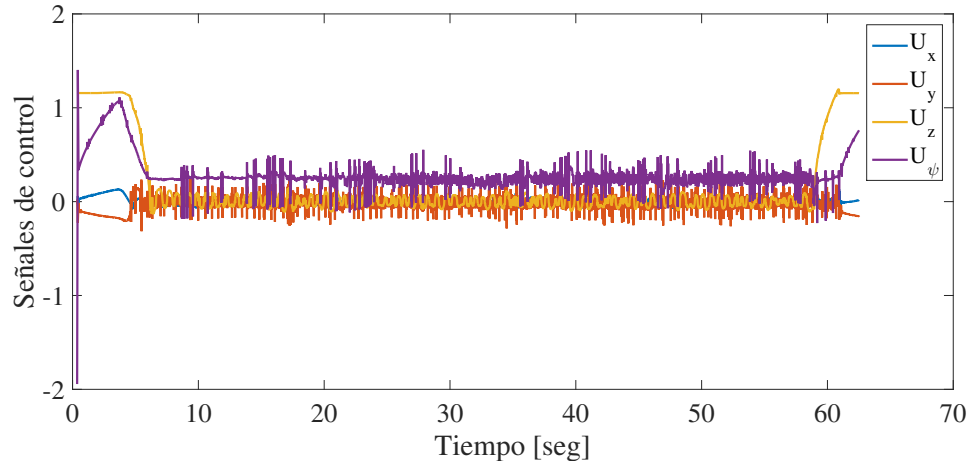


Figura 6.6: Señales de control del cuadrirrotor para la trayectoria circular.

Es posible observar señales de control de una magnitud relativamente mayor al inicio y final del experimento. Esto es debido a que el algoritmo matemático comienza a

actuar justo en el despegue del cuadrirrotor, y continua actuando hasta el aterrizaje; sin embargo, las referencias continúan generando valores de manera normal (como se muestra en la Figura 6.4), por lo que el algoritmo de control trata de seguirlas. En el caso del despegue, esto sucede debido a la gran magnitud de error que se tiene justo en el instante que se está en reposo. Para el caso del aterrizaje, es debido a que este proceso es independiente del control de posición, por lo que sigue actuando mientras el cuadrirrotor se dirige a *tierra*.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: <https://youtu.be/CAZUM3N0kcg>

6.1.2 Trayectoria en espiral

En este experimento, se realiza una trayectoria en espiral; esto es, realizando una elipse en el plano XY , mientras la altura va cambiando uniformemente. Para la realización de la elipse se utiliza una ecuación similar a 6.1; sin embargo, se agrega un factor de alargamiento e_e en uno de los ejes, para formar la elipse por afinidad de los diámetros formados por las ecuaciones x y y . Estas son:

$$\begin{aligned} x_e &= x_{centroide_e} + r \sin(2\pi f_e t) \\ y_e &= y_{centroide_e} + e_e r \cos(2\pi f_e t) \end{aligned} \tag{6.2}$$

donde x_e y y_e representan a las coordenadas en el marco inercial fijo de la trayectoria. Además, $x_{centroide_e}$ y $y_{centroide_e}$ corresponden a las coordenadas donde está el centro de la elipse descrita en el plano XY , mientras que f_e , t y r corresponden a las *revoluciones por segundo*, al tiempo y al radio principal, respectivamente.

Para el caso de la variación de la altura, se definen la altura inicial y la altura final del espiral, para posteriormente construir una función lineal de *la recta* y así variar linealmente la altura calculando una *pendiente* con respecto al *tiempo* en que se desea que se alcance la altura máxima. Esto se realiza por medio de la ecuación (6.3).

$$z_e = \frac{h_{fe} - h_{ie}}{t_f} t + h_{ie} \quad (6.3)$$

donde z_e es la coordenada de altura en el marco inercial fijo. La altura final es representada como h_{fe} mientras que la altura inicial es h_{ie} . El tiempo final en que se desea que se alcance la altura final es definido como t_f y t es el tiempo.

En este experimento, los siguientes datos son utilizados:

- Radios de la elipse con un factor de alargamiento de 1.5:
 - Radio menor de la elipse descripta de 1500 *mm*.
 - Radio mayor de la elipse descripta de 2200 *mm*.
- Coordenadas del centro de la elipse en:
 - $x_{centroide} = 0$ *mm*.
 - $y_{centroide} = 0$ *mm*.
- Velocidad de 0.04 *rev/seg* (2.4 *rpm*).
- Alturas del espiral:
 - Altura inicial de 1000 *mm*.
 - Altura final de 3500 *mm*.
- Tiempo de 60 *seg* en alcanzar la altura final.

Entonces, utilizando los datos de las ecuaciones (6.2) y (6.3) como referencia de posición, así como el control de orientación propuesto en (3.27), se obtiene el seguimiento del espiral de referencia, ver Figura 6.7, donde las líneas que van hacia *tierra* representan el aterrizaje y despegue del cuadrirrotor.

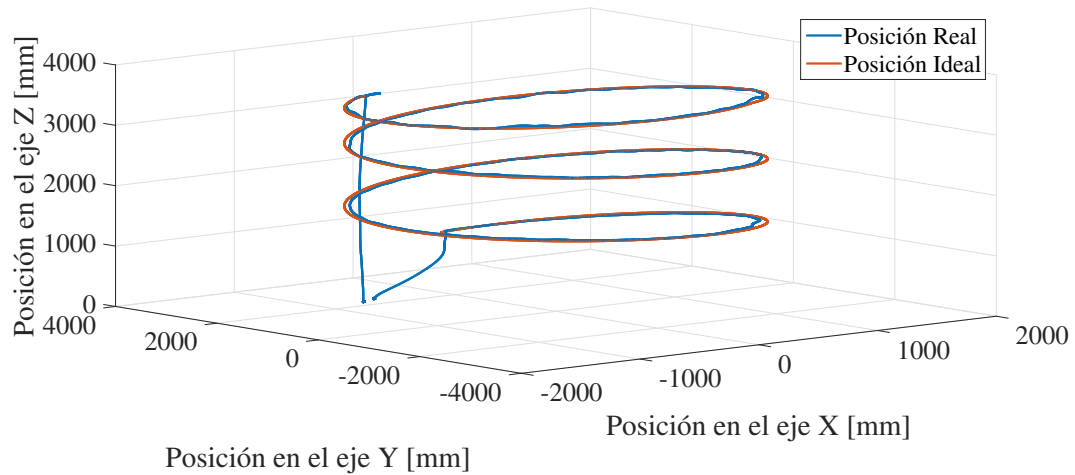


Figura 6.7: Trayectoria en espiral en el espacio tridimensional.

Además, en la Figura 6.8, se observa la evolución del ángulo de *guiñada* con respecto a su referencia tangente a la trayectoria de la elipse formada por el espiral en el plano XY .

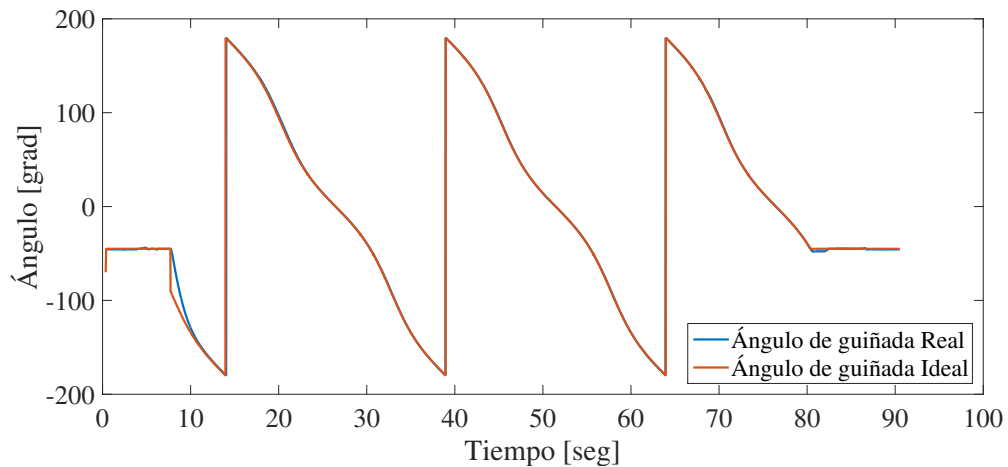
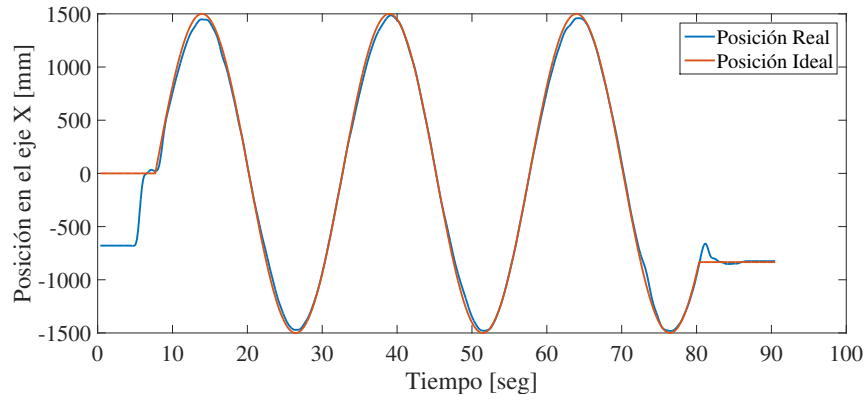
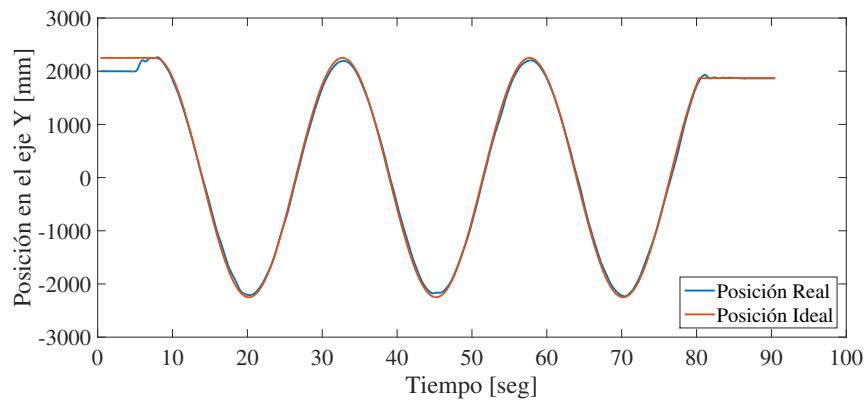


Figura 6.8: Ángulo de guiñada del cuádrirrotor siguiendo el espiral.

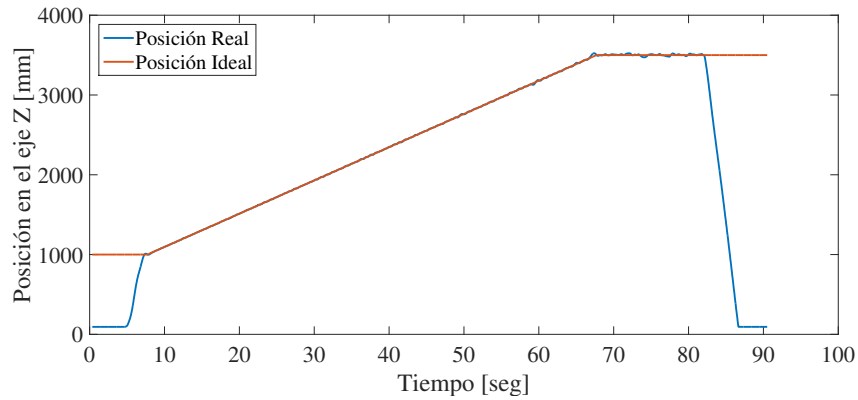
En la Figura 6.9 se muestran los resultados individuales del seguimiento de la trayectoria para cada eje de movimiento del cuádrirrotor.



(a) Posición en el eje X.



(b) Posición en el eje Y.



(c) Posición en el eje Z.

Figura 6.9: Gráficas de posiciones XYZ por eje para la trayectoria en espiral.

Utilizando las ecuaciones (3.14) y (3.26), se generaron las *Superficies deslizantes* correspondientes y se muestran en la Figura 6.10.

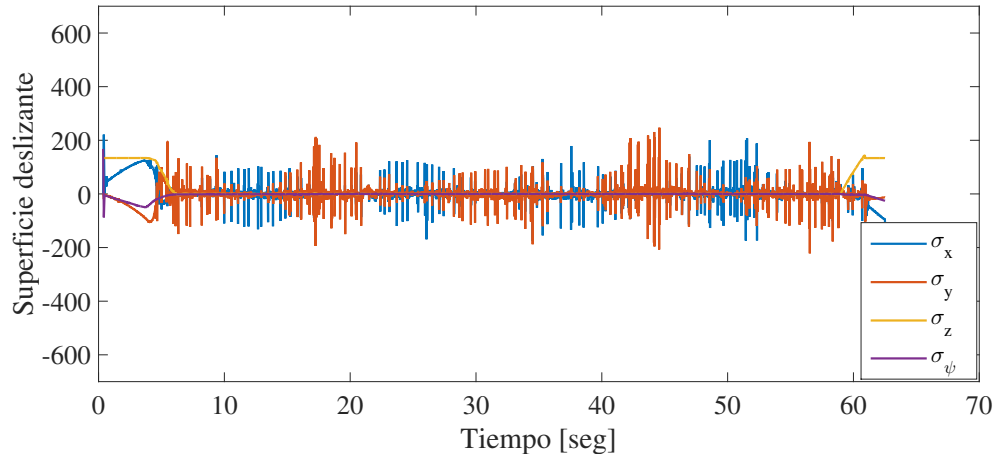


Figura 6.10: Superficies deslizantes generadas utilizadas para el control en la trayectoria en espiral.

Con las superficies deslizantes mostradas, se calcula el control por medio del algoritmo de *Super Twisting* resultando en las señales de control mostradas en la Figura 6.11.

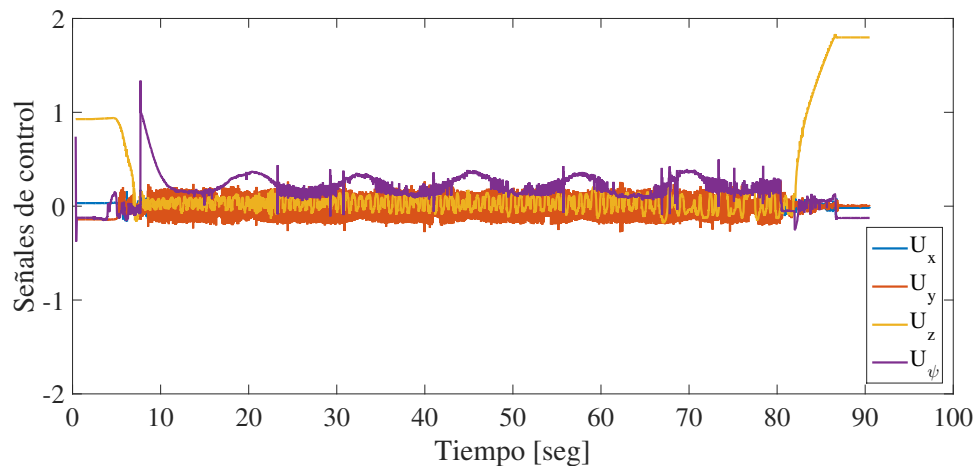


Figura 6.11: Señales de control del cuadrirrotor para la trayectoria en espiral.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: <https://youtu.be/Ffr3JcIIERY>

6.1.3 Respuesta a perturbaciones físicas

Para probar la resistencia del control *Super Twisting* frente a perturbaciones del entorno, se realiza una prueba empírica para poder observar el comportamiento del cuadrirrotor.

El experimento realizado consiste en la aplicación de una fuerza no medida sobre el cuerpo del cuadrirrotor; esto es, empujándolo de tal manera que altere considerablemente su posición en el espacio coordenado. Esto se realiza mientras el vehículo está estático realizando *vuelo estacionario*²³, y solo en el eje Y del marco inercial. ver Figura 6.12.

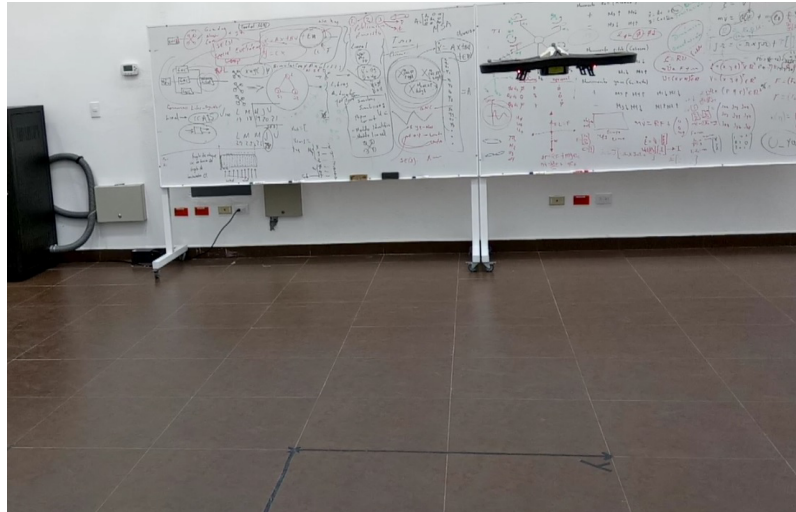


Figura 6.12: Cuadrirrotor en *vuelo estacionario*.

Habiendo empujado el cuadrirrotor, se alcanza una distancia máxima la cual es el error con respecto al punto de referencia que sigue constante. Esto puede ser visto en la Figura 6.13.

²Más conocido por el término en idioma inglés *hover*

³Además, el experimento se realiza teniendo el cuadrirrotor con un ángulo de guiñada de 90° por motivos de simplicidad al aplicar la perturbación



Figura 6.13: Cuadrirotor después de la perturbación.

Posteriormente, la acción de control regresa el cuadrirotor a su referencia original, pudiéndose percibir un *sobre-tiro* debido a la inercia del movimiento; sin embargo, su magnitud es mínima en comparación con el error inducido por la perturbación.

En la Figura 6.14 se muestran los datos de posición durante la realización del experimento.

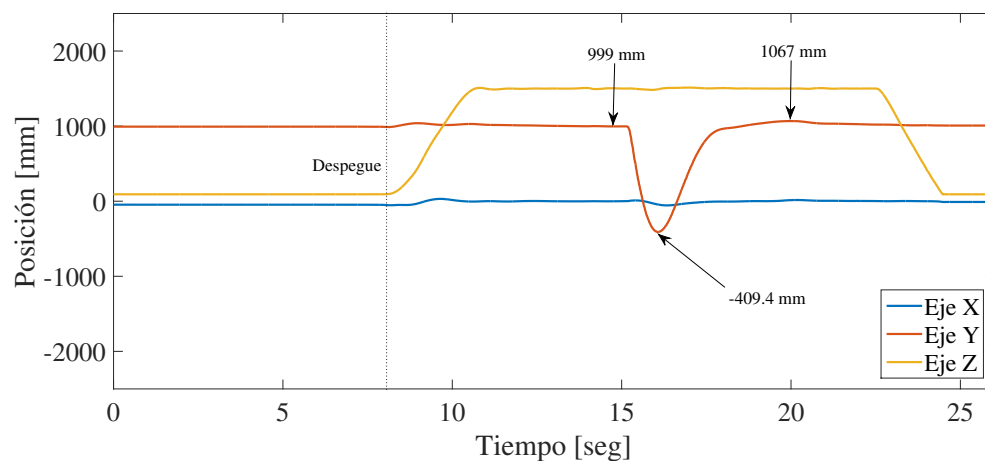


Figura 6.14: Datos de posición durante el experimento de perturbación.

Teniendo en cuenta los datos anteriores, es posible resumir los resultados de la siguiente manera:

- Coordenadas ideales del *vuelo estacionario*:
 - Coordenada X : 0 mm.
 - Coordenada Y : 1000 mm.
- Posición máxima alcanzada debido a la perturbación en el eje Y : -409.4 mm.
- Error de posición máximo inducido: 1408.4 mm.
- Sobre-tiro máximo registrado: 68 mm.

En la Figura 6.15 se muestra la velocidad en el eje Y alcanzada por el cuadrirrotor debido a la perturbación. Es posible notar una velocidad negativa la cual corresponde al momento en que el cuadrirrotor avanza hacia su error máximo, posteriormente la velocidad tiende a cero por un instante, correspondiendo al cambio de sentido en el movimiento, y luego comienza el retroceso hacia la referencia.

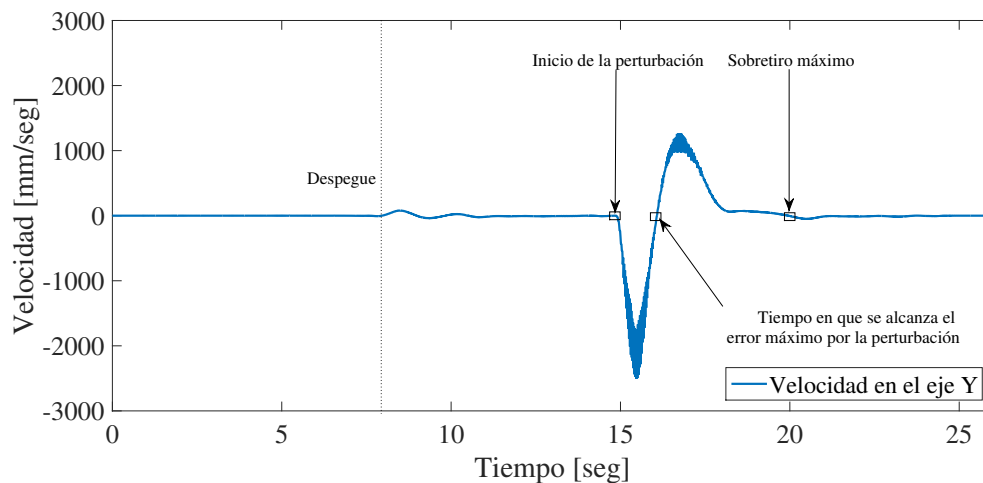


Figura 6.15: Velocidad en el eje Y debido a la perturbación inducida.

Posteriormente, en la Figura 6.16 se observan los ángulos de alabeo y cabeceo del cuadrirrotor durante la realización del experimento. Es posible notar un incremento

súbito en el ángulo de alabeo debido al controlador compensando la perturbación y forzando al vehículo a regresar a su posición de referencia.

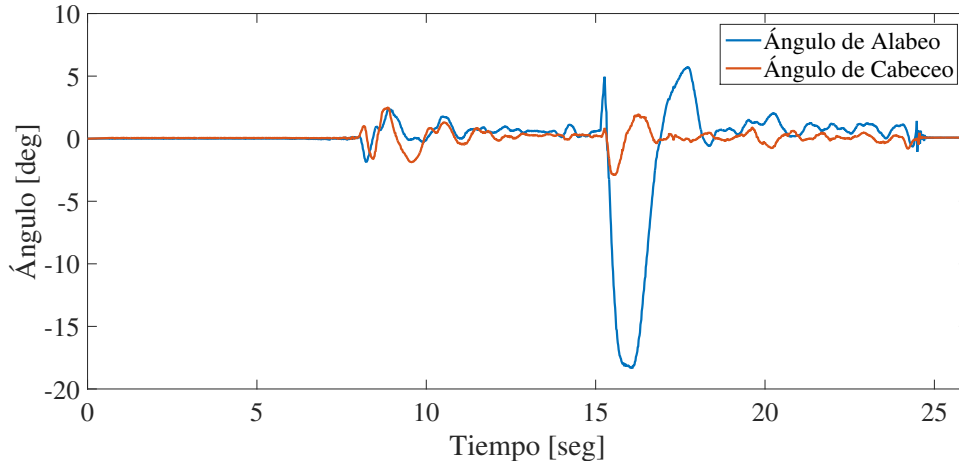


Figura 6.16: Ángulos de alabeo y cabeceo del cuadrirrotor durante la perturbación.

En la Figura 6.17 se muestran las superficies deslizantes generadas por el control, donde es posible observar la alteración de σ_y debido a que fue el eje del marco fijo donde ocurrió principalmente la perturbación.

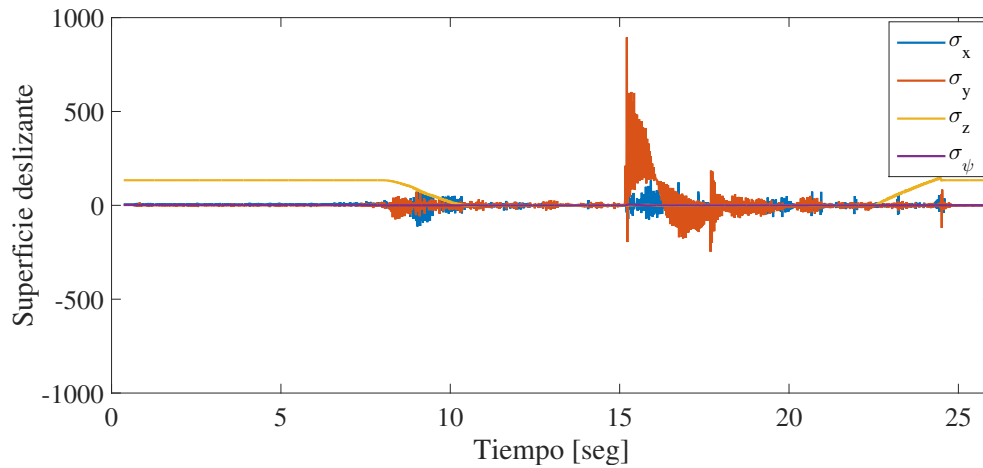


Figura 6.17: Superficies deslizantes producidas durante la perturbación.

Finalmente, en la Figura 6.18 se grafican las señales de control generadas por el algoritmo de *Super Twisting*. A diferencia de de las superficies deslizantes, la señal

de control principalmente afectada por la perturbación es U_x debido a que el experimento se realizó con el cuadrirrotor rotado 90° en el eje Z .

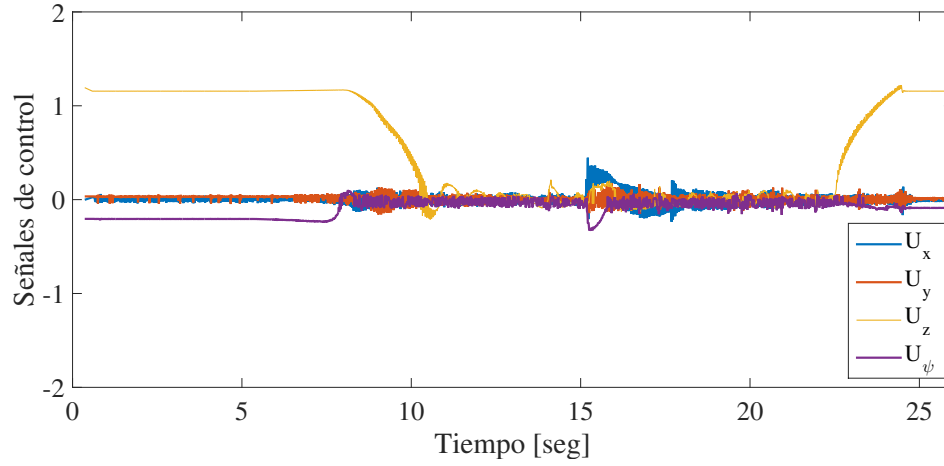


Figura 6.18: Señales de control producidas durante la perturbación.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: <https://youtu.be/nIIcxWLOAXc>

6.2 *Consensus* de segundo orden

Como primera aproximación para el vuelo en formación de un sistema multi-agentes, se opta por la utilización de un algoritmo de coordinación basado en el esquema de *Consensus* para dinámicas de segundo orden expuesto en la sección 4.4.1.

La arquitectura principal de este arreglo consiste en un *Algoritmo Gestor* el cual alimenta los respectivos *Controles de Navegación Locales* de cada agente; esto es, que la referencia deseada para los controladores no proviene directamente de la trayectoria deseada, si no del gestor que se encarga de coordinar la posición y velocidad de los agentes dependiendo de los estados del sistema [24].

Para establecer lo anterior, se define la dinámica de movimiento del i -ésimo agente como:

$$\dot{\lambda}_i = A\lambda_i + Bu_{c_i} \quad (6.4)$$

$$\zeta_i = C\lambda_i \quad (6.5)$$

donde $\lambda_i \in \mathbb{R}^{1 \times p}$ corresponde al vector de estados del i -ésimo agente del sistema, $\zeta_i \in \mathbb{R}^{1 \times p}$ es el vector de salida medido del i -ésimo agente, y A , B y C son matrices constantes de dimensiones compatibles que caracterizan la dinámica del agente.

DEFINICIÓN 6.1 *Un sistema multi-agentes alcanza el consensus si para cada agente $i = \{1, \dots, N\}$ existe una entrada de control con retroalimentación de estado de tal manera que el lazo cerrado del sistema satisfaga que:*

$$\lim_{t \rightarrow \infty} \|(\zeta_i(t) - h_i(t)) - \lambda_i(0)\| = 0 \quad (6.6)$$

con condiciones iniciales $\lambda_i(0)$ y donde $h_i(t) \in \mathbb{R}^{1 \times m}$ representa el vector de formación; esto es, las posiciones y velocidades deseadas, correspondientes a la trayectoria a seguir definidas como $h_i(t) = [h_{X_i}, h_{Y_i}]^\top$.

Tomando en cuenta la Definición 6.1, se puede establecer el protocolo *consensus* como:

$$u_{c_i}(t) = K\zeta_i(t) + K_g\tilde{\zeta}_i(t) + K_f \sum_{j \in N_i} a_{ij}(\tilde{\zeta}_j(t) - \tilde{\zeta}_i(t)) + \dot{h}_i(t) \quad (6.7)$$

donde es posible identificar las siguientes partes:

- N_i es el conjunto de vecinos para el agente i .
- $\zeta_i(t) = [x(t), \dot{x}(t), y(t), \dot{y}(t)]^\top$ corresponde al vector de estados⁴ del agente i .
- $\tilde{\zeta}_i(t) = \zeta_i(t) - h_i(t)$ es el vector de errores de salida para el agente i .
- $\tilde{\zeta}_j(t) = \zeta_j(t) - h_j(t)$ es el vector de errores de salida para el agente vecino j .

⁴Solo se toman los estados de los ejes X y Y debido a que es donde se realiza el *consensus*

- $\dot{h}_i(t)$ corresponde a un término de compensación [25],[26].
- $K_g = [k_{p_g} \ k_{v_g}]$ son las ganancias que garantizan la convergencia de los estados del agente i a la trayectoria deseada h_i ; esto es, $\zeta_i \rightarrow h_i$.
- $K_f = [k_{p_f} \ k_{v_f}]$ son las ganancias que garantizan que los agentes lleguen al *consensus* en sus vectores de error $\tilde{\zeta}_i$ y $\tilde{\zeta}_j$.
- $K = [k_1 \ k_2]$ corresponden a las ganancias para mantener la posición y velocidad en lazo cerrado del agente i .

Además, a diferencia del protocolo visto en (4.1), el *consensus* se realiza en los *errores* de los agentes, lo que significa que siempre buscarán tener un error igual, por lo que si uno se desvía, el otro lo seguirá, independientemente de la trayectoria asignada por separado a cada agente.

La arquitectura o arreglo principal que se utiliza para la realización de los experimentos se muestra en la Figura 6.19.

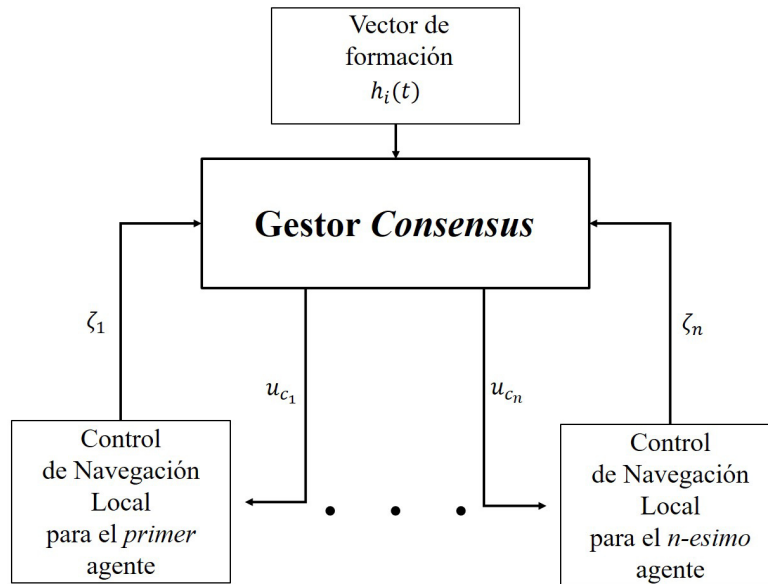


Figura 6.19: Arquitectura del sistema de navegación por *Consensus* de segundo orden.

Así mismo, el grafo de conectividad utilizado se muestra en la Figura 6.20. Se puede observar que existe un flujo de información continuo entre agentes debido a que el último vuelve a comunicar la información con el primero.

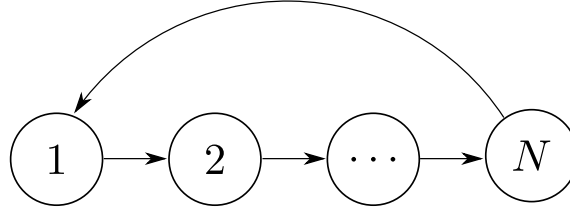


Figura 6.20: Grafo de conexión utilizado para los experimentos de *Consensus* de segundo orden

Finalmente, la *matriz de adyacencia* A y la *matriz laplaciana* L para el grafo mostrado en la Figura 6.20, y tomando en cuenta dos agentes, se muestran a continuación:

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad L = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

6.2.1 Trayectoria circular con dos agentes (con perturbación)

Para la realización de este experimento, se utilizan dos cuadrirrotores, y de manera similar que en la sección 6.1.1, se utilizan las ecuaciones paramétricas de un círculo, (6.1), para generar las coordenadas de la trayectoria, tendiendo como única diferencia el adelanto de fase de un agente, para evitar el seguimiento de un mismo punto. Los datos de la trayectoria utilizada en este experimento son:

- Radio del círculo de 1500 *mm*.
- Coordenadas del centro del círculo en el origen en ambos ejes.
- Velocidad de 0.04 *rev/seg* (2.4 *rpm*).

- Ángulo de separación entre agentes de 180° .
- Distancia lineal entre agentes de 3000 mm .

Además, para visualizar el efecto del algoritmo *consensus*, se perturba cada agente de la misma manera que en 6.1.3. Los resultados del seguimiento son mostrados en la Figura 6.21.

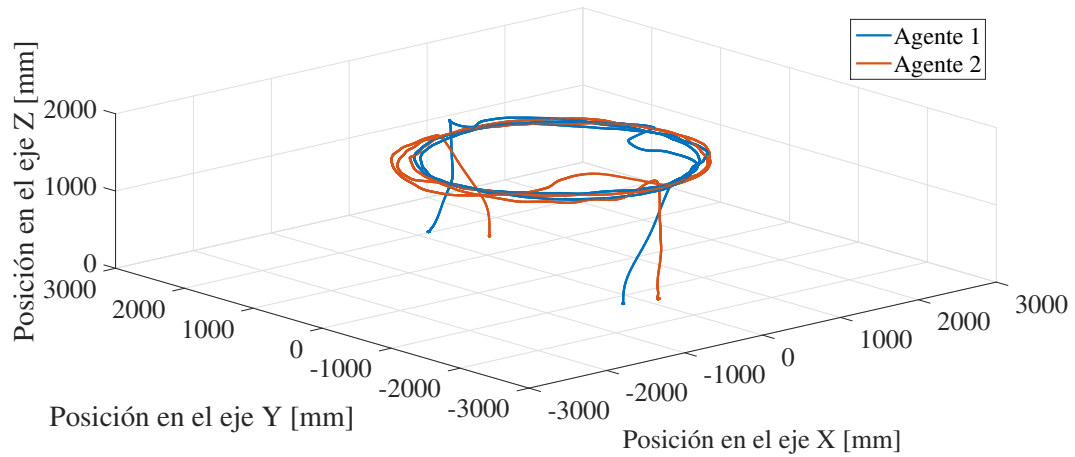


Figura 6.21: Trayectoria circular con *Consensus* de segundo orden en el espacio tridimensional.

Los ángulos de *guiñada* tangentes a la trayectoria se muestran en la Figuras 6.22 y 6.23, respectivamente.

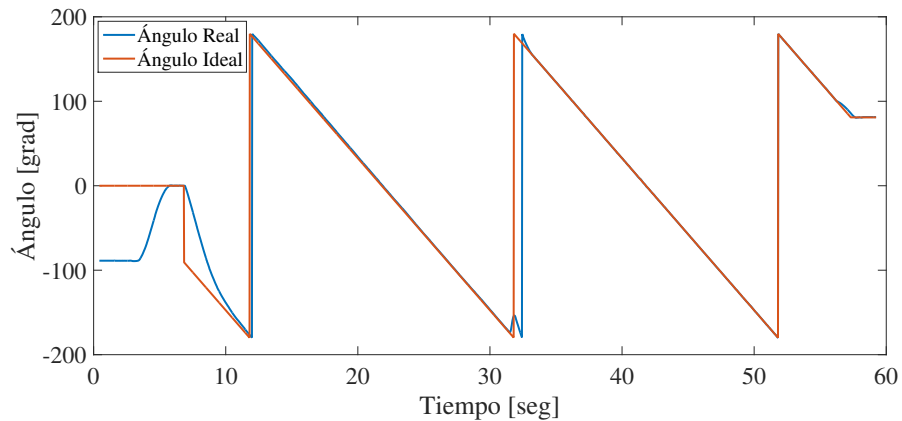


Figura 6.22: Ángulo de guiñada para el Agente 1.

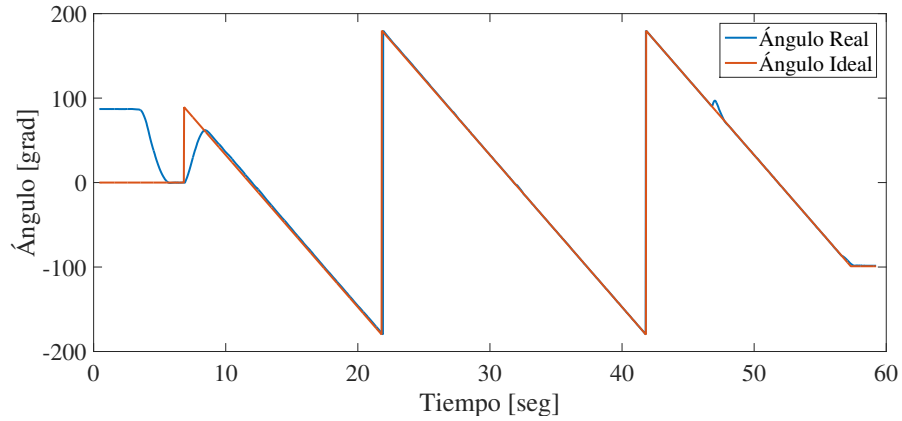


Figura 6.23: Ángulo de guiñada para el Agente 2.

Las posiciones en los ejes por separado para el Agente 1 se muestran en las Figuras 6.24, 6.25 y 6.26. Es posible observar la acción del algoritmo *consensus* de manera más clara en estas gráficas.

En la Figura 6.24 se puede observar como se perturba el Agente 1, y en la Figura 6.25 se puede notar como la perturbación hecha en el Agente 2 afecta al Agente 1 forzándolo a mantener la formación y a seguirlo.

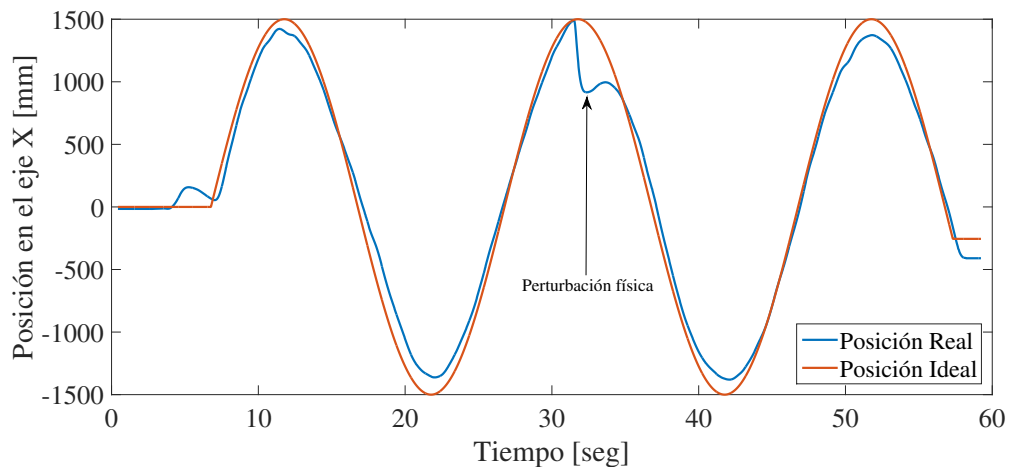
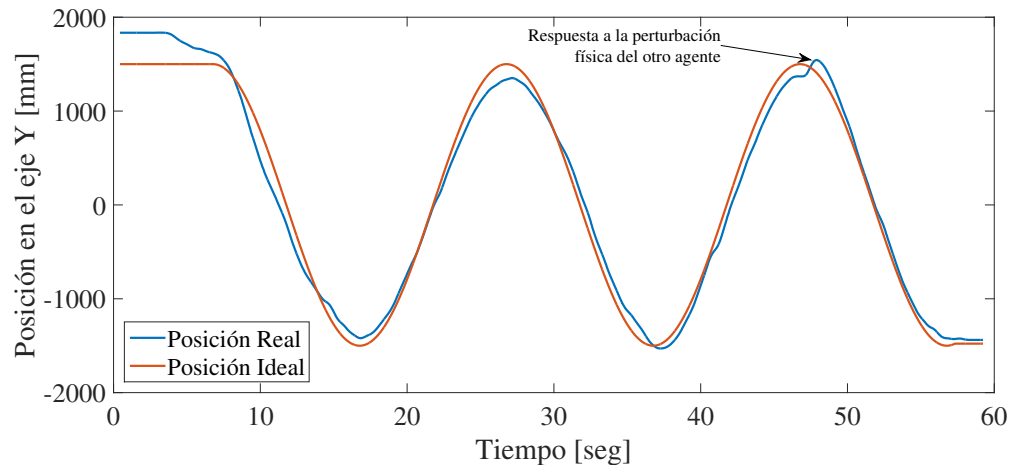
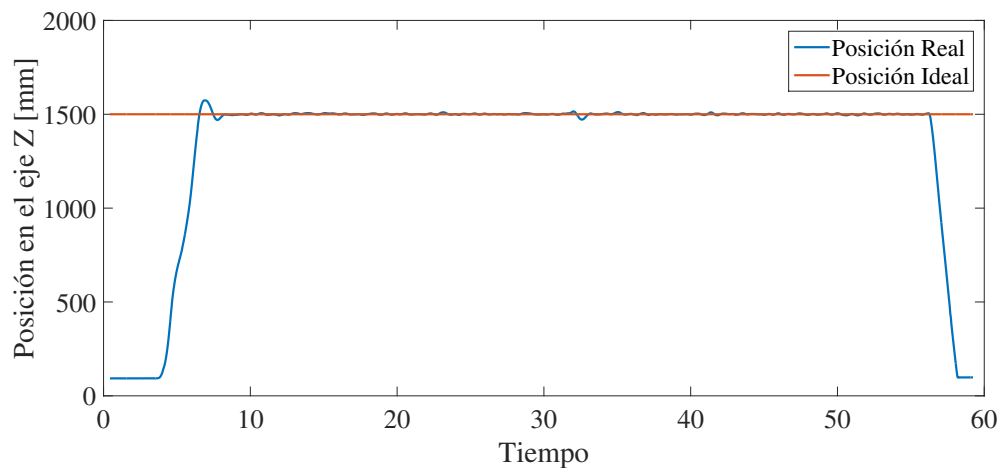
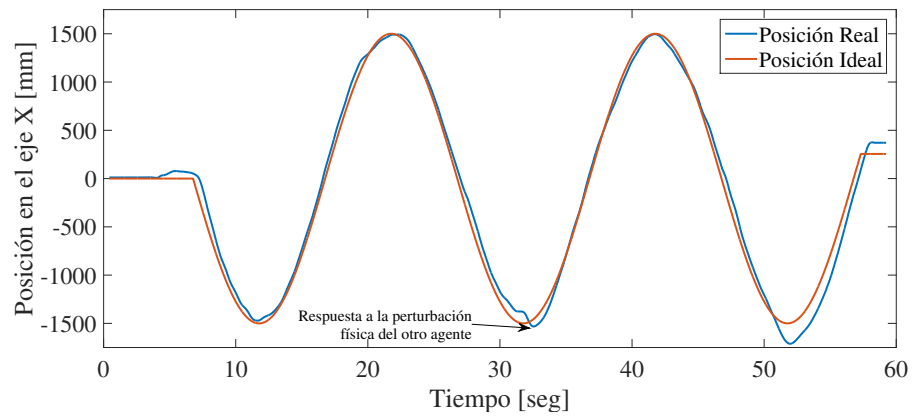
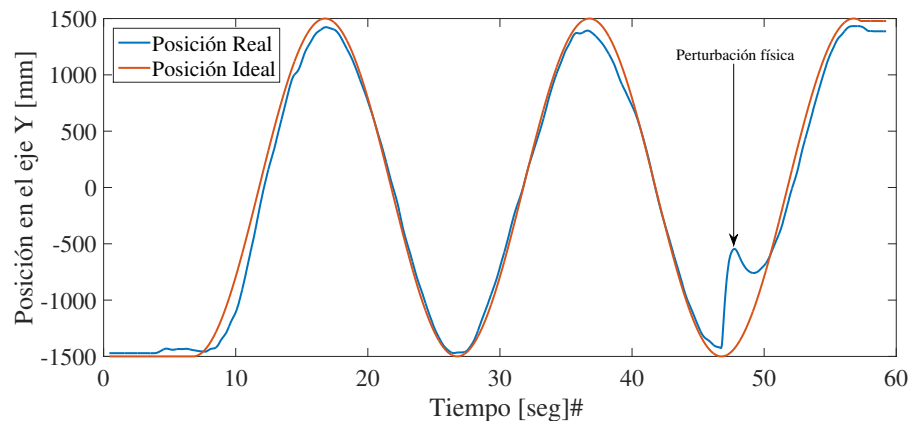
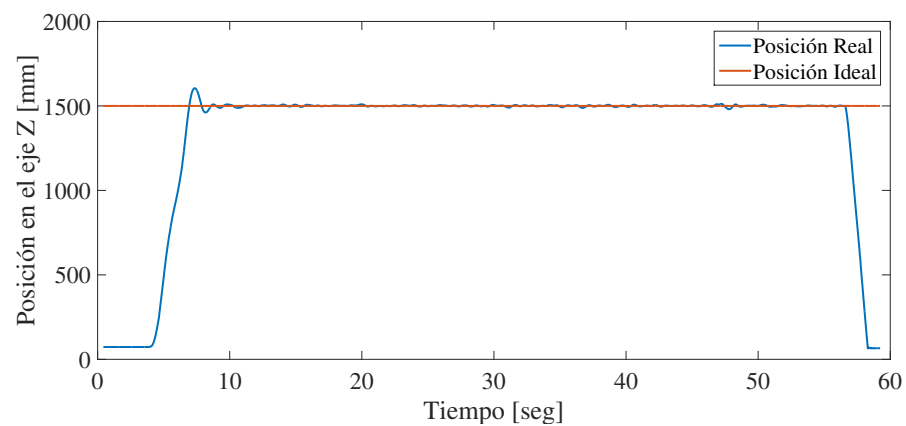


Figura 6.24: Posiciones en el eje X para el Agente 1.

Figura 6.25: Posiciones en el eje Y para el Agente 1.Figura 6.26: Posiciones en el eje Z para el Agente 1.

De la misma manera, en las Figuras 6.27, 6.28 y 6.29 se muestran los datos de seguimiento para cada eje del Agente 2. De igual manera, es posible observar en la Figura 6.27 como la perturbación en el Agente 1 afecta al Agente 2 y hace que lo siga. Además, en la Figura 6.28 se puede notar la perturbación hecha directamente sobre el Agente 2.

Figura 6.27: Posiciones en el eje X para el Agente 2.Figura 6.28: Posiciones en el eje Y para el Agente 2.Figura 6.29: Posiciones en el eje Z para el Agente 2.

Los resultados anteriores son logrados por medio de las entradas de referencia generadas por el *Gestor Consensus* el cual, haciendo uso de la trayectoria deseada y los estados de los agentes, produce los datos mostrados en la Figura 6.30 para el eje X y en la Figura 6.31 para el eje Y .

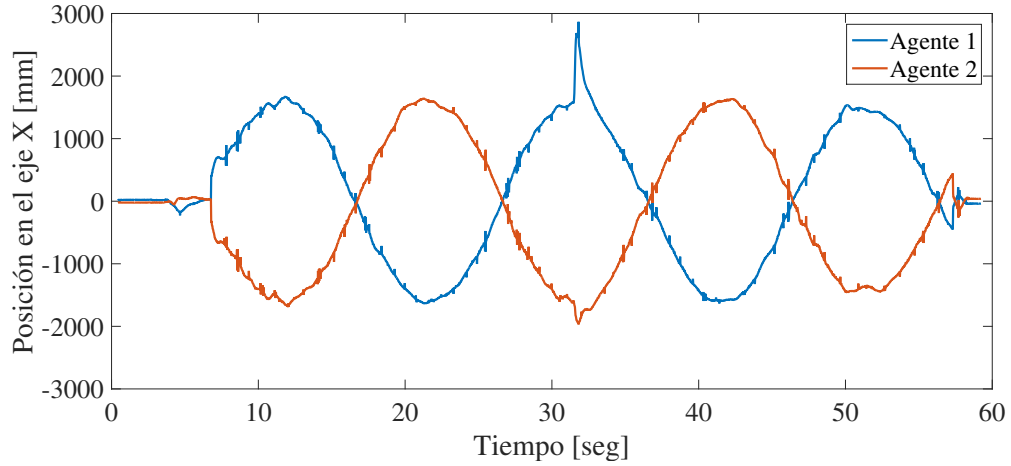


Figura 6.30: Señales generadas por el *Gestor Consensus* para el eje X .

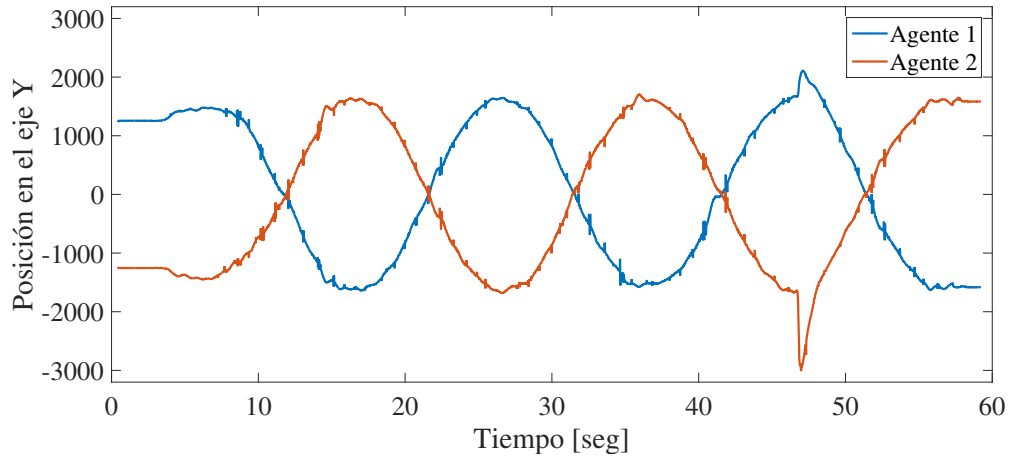


Figura 6.31: Señales generadas por el *Gestor Consensus* para el eje Y .

Utilizando estas referencias como entrada para los respectivos controladores *Super Twisting* se obtienen las superficies deslizantes correspondientes mostradas en las Figuras 6.32 y 6.33 para el Agente 1 y el Agente 2, respectivamente.

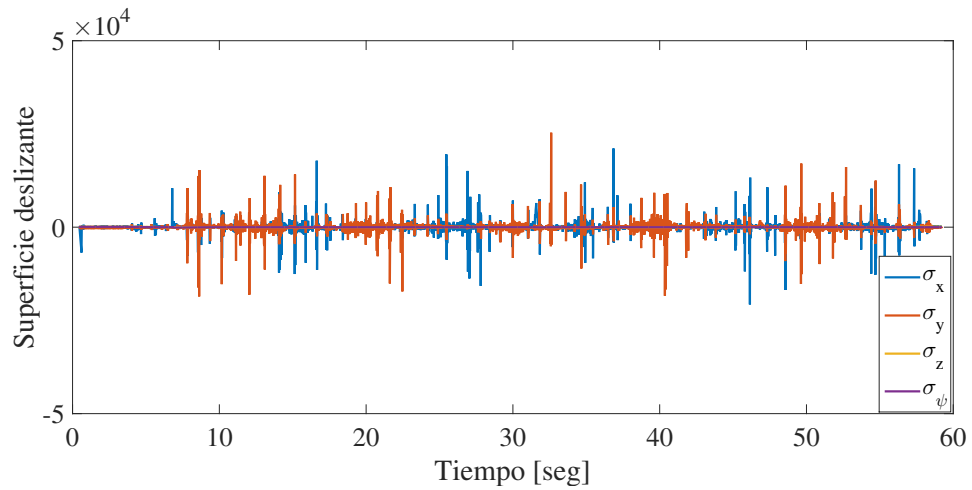


Figura 6.32: Superficies deslizantes del Agente 1.

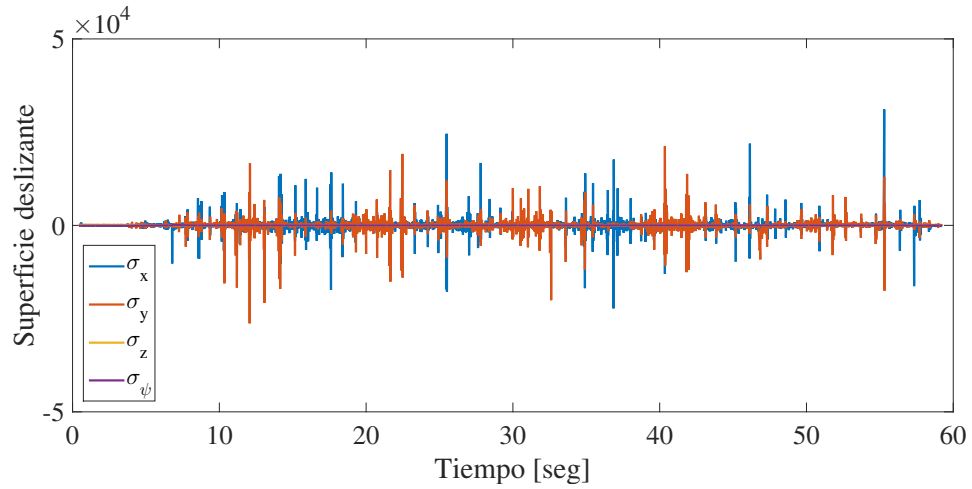


Figura 6.33: Superficies deslizantes del Agente 2.

Finalmente, con las superficies deslizantes mostradas, se generan las señales de control para cada agente. Las señales de control para el Agente 1 se muestran en la Figura 6.34 y para el Agente 2 en la Figura 6.35.

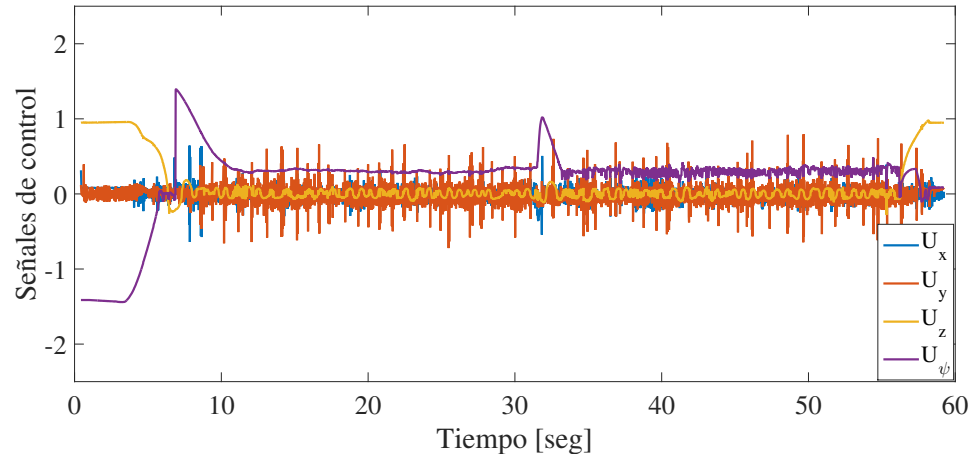


Figura 6.34: Señales de control para el Agente 1.

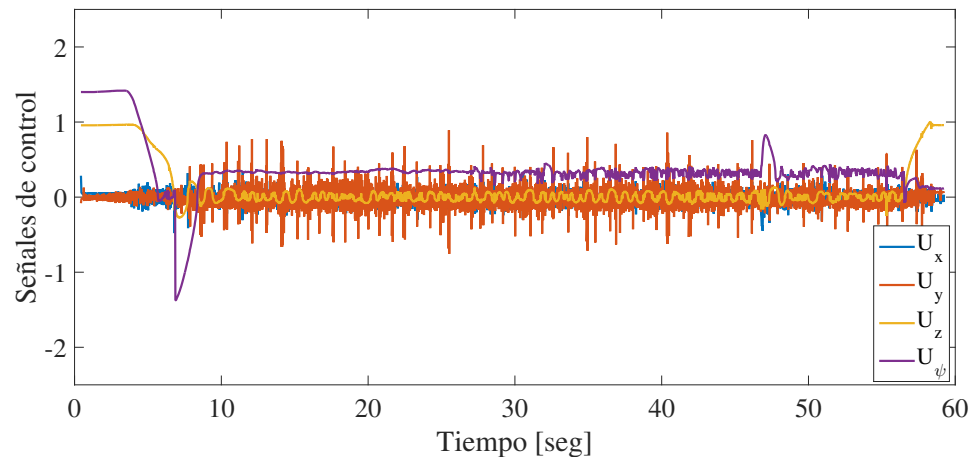


Figura 6.35: Señales de control para el Agente 2.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: <https://youtu.be/7TM9PK1xEZc>

6.2.2 Trayectoria para cubrir un área cuadrada

En este experimento se realiza una trayectoria con dos agentes para cubrir un área cuadrada de manera más eficiente que si se realizara con uno solo; esto es, con un agente se tendrían que realizar más vueltas y llevaría más tiempo que utilizando dos.

La trayectoria deseada se muestra en la Figura 6.36, donde se observa como los agentes realizan una serie de vueltas para cubrir el área aprovechando la presencia de ambos.

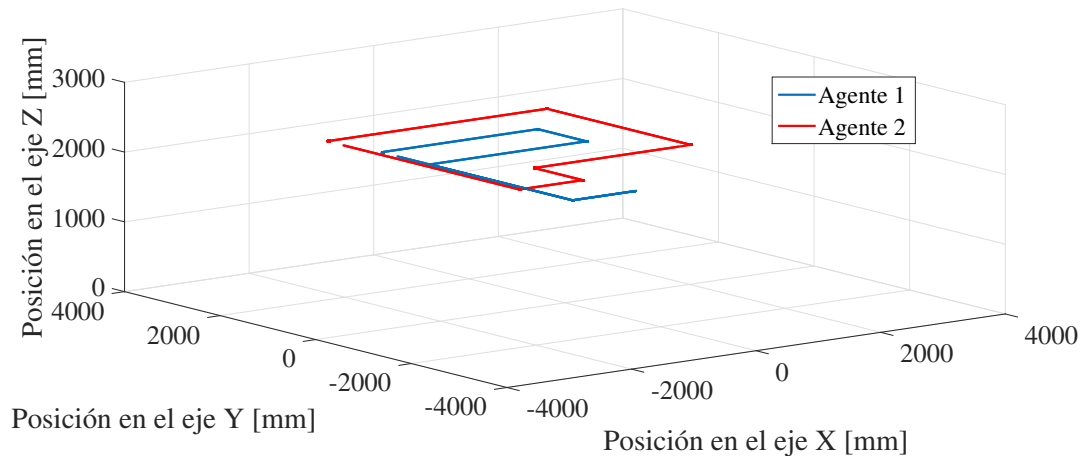


Figura 6.36: Trayectoria deseada para cubrir un área cuadrada.

Los resultados del seguimiento experimental utilizando dos agentes se muestran en la Figura 6.37.

Además, la guiñada de ambos agentes sigue tangencialmente la trayectoria, cambiando de valor solo cuando ambos han llegado a su punto de giro. Esto puede ser visto en las Figuras 6.38 y 6.39 donde se observa el ángulo de guiñada ideal contra el ángulo de guiñada descrito por cada uno de los agentes.

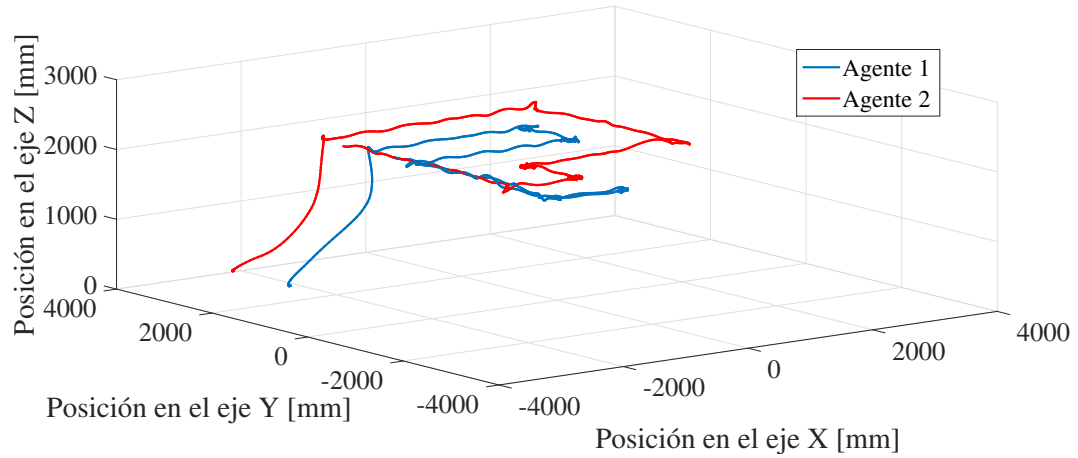


Figura 6.37: Seguimiento real para cubrir un área cuadrada.

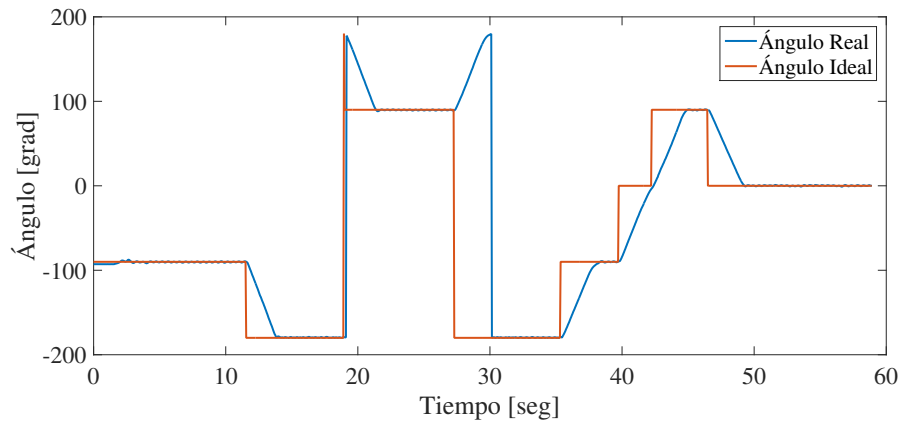


Figura 6.38: Ángulo de guiñada del Agente 1.

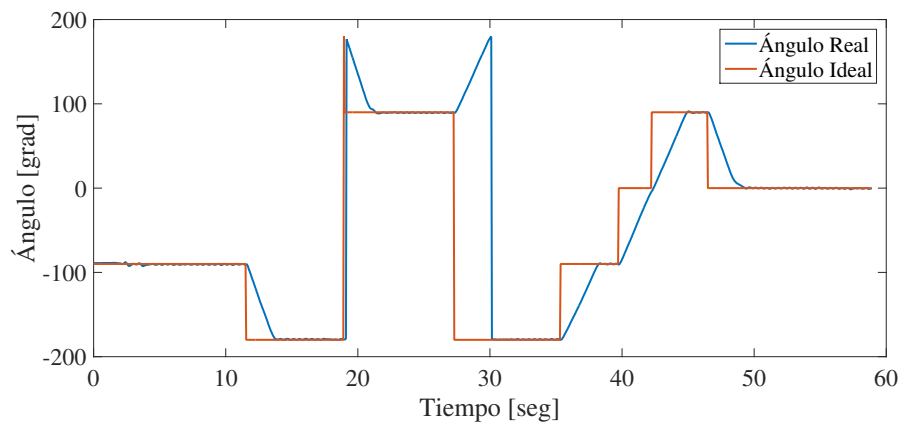
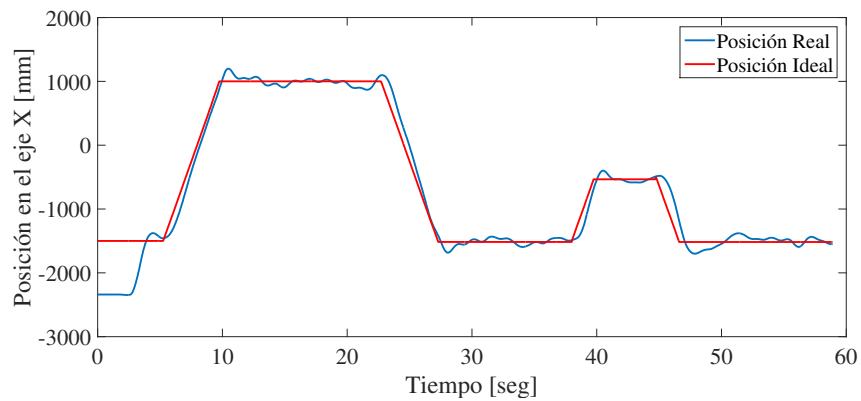
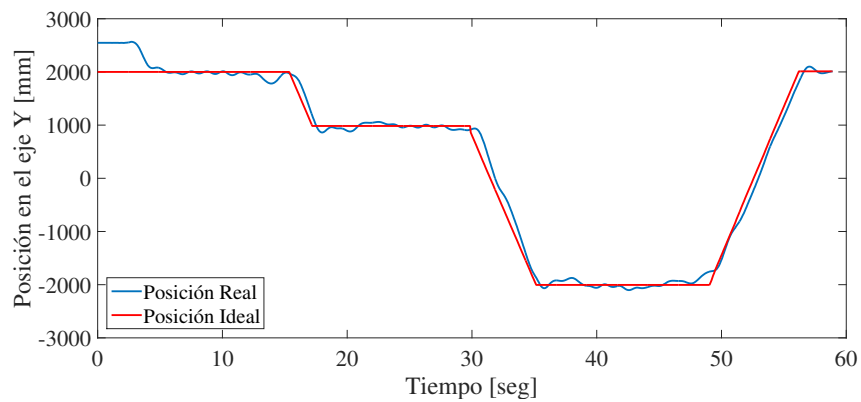


Figura 6.39: Ángulo de guiñada del Agente 2.

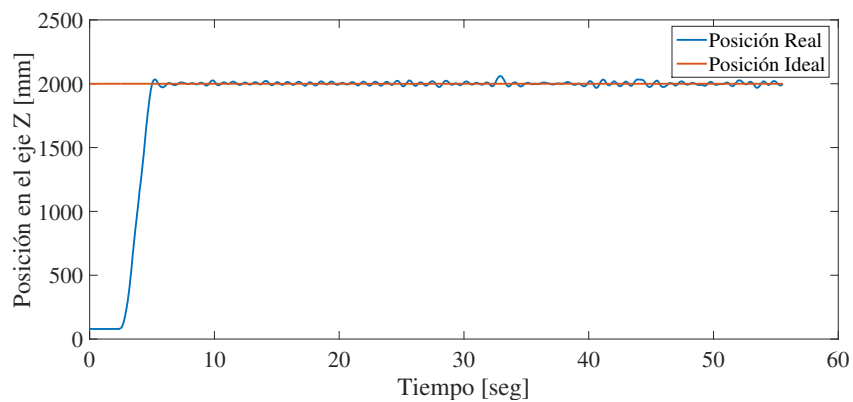
De igual manera en la Figura 6.40 se presentan las gráficas de seguimiento para los tres ejes coordenados de manera individual para el Agente 1.



(a) Posición en el eje X.



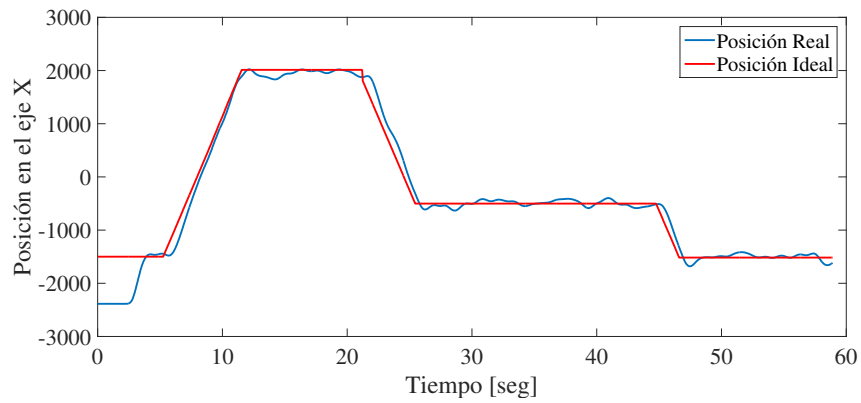
(b) Posición en el eje Y.



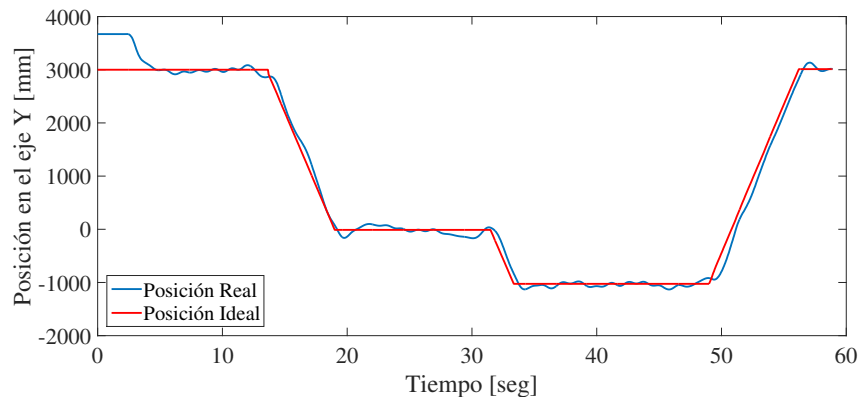
(c) Posición en el eje Z.

Figura 6.40: Gráficas de posiciones XYZ por eje para la trayectoria para cubrir un área cuadrada del Agente 1.

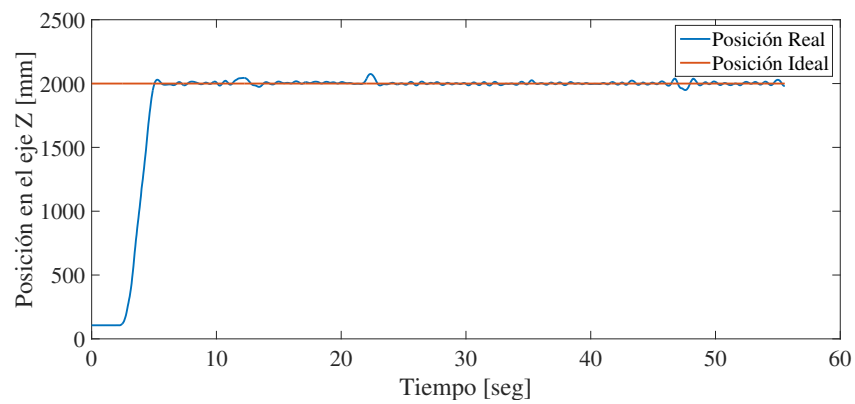
Para el Agente 2, las gráficas para los ejes coordenados son mostradas en la Figura 6.41.



(a) Posición en el eje X.



(b) Posición en el eje Y.



(c) Posición en el eje Z.

Figura 6.41: Gráficas de posiciones XYZ por eje para la trayectoria para cubrir un área cuadrada del Agente 2.

Las posiciones generadas por el *Gestor Consensus* que son tomadas como entradas por el control son mostradas en las Figuras 6.42 y 6.43.

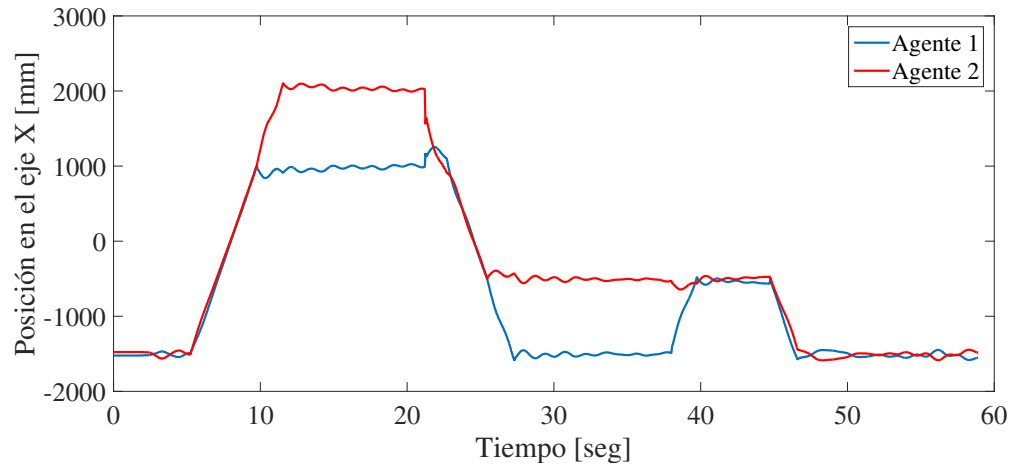


Figura 6.42: Señales generadas por el *Gestor Consensus* para el eje X.

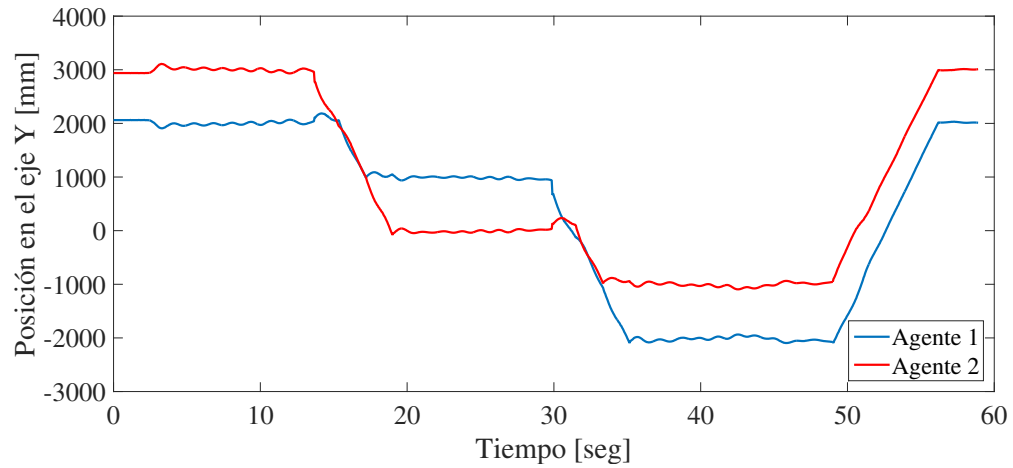


Figura 6.43: Señales generadas por el *Gestor Consensus* para el eje Y.

Tomando estas señales como referencia para los controladores, las superficies deslizando son generadas para cada agente como se muestra en las Figuras 6.44 y 6.45.

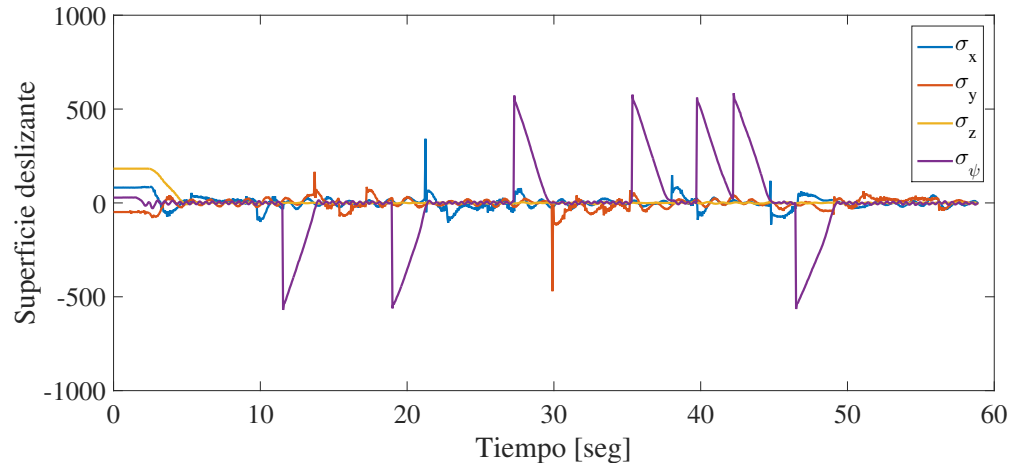


Figura 6.44: Superficies deslizantes del Agente 1.

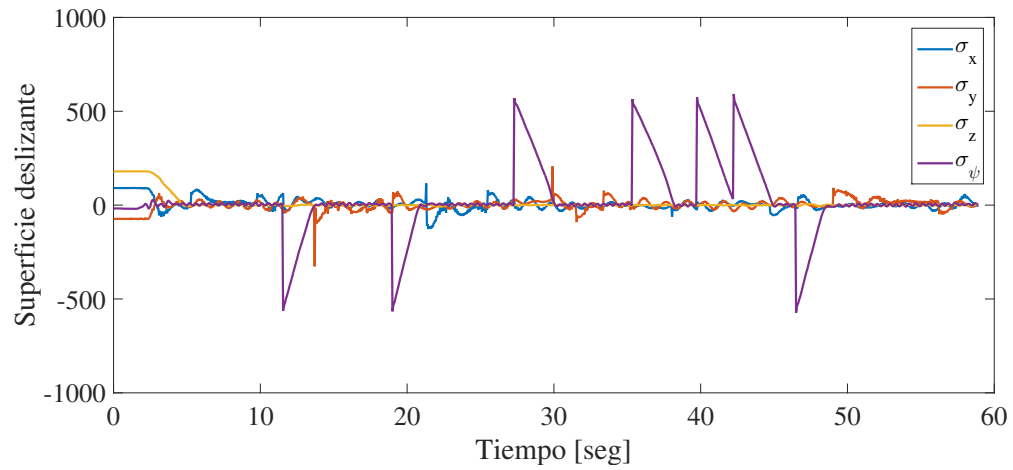


Figura 6.45: Superficies deslizantes del Agente 2.

Haciendo uso de estas superficies deslizantes, se generan las distintas señales de control correspondientes para el cuádrirrotor. Estas señales son mostradas en las Figuras 6.46 y 6.47 para el Agente 1 y el Agente 2, respectivamente.

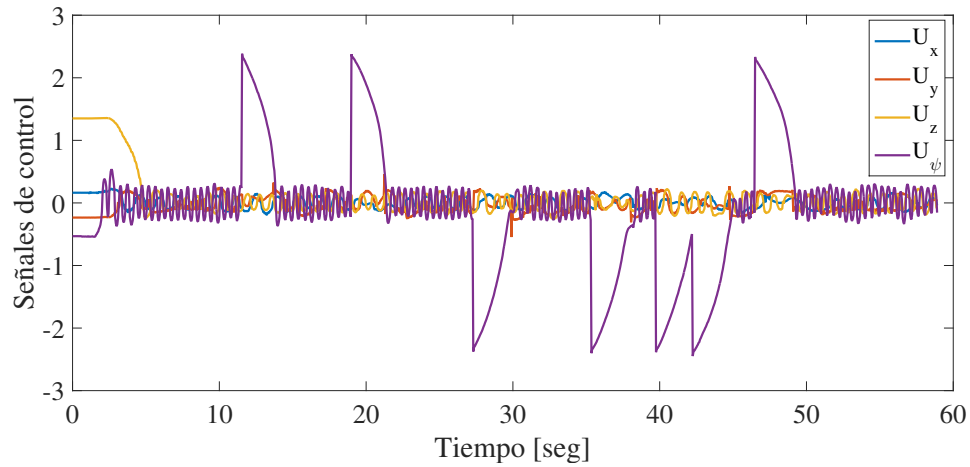


Figura 6.46: Señales de control para el Agente 1.

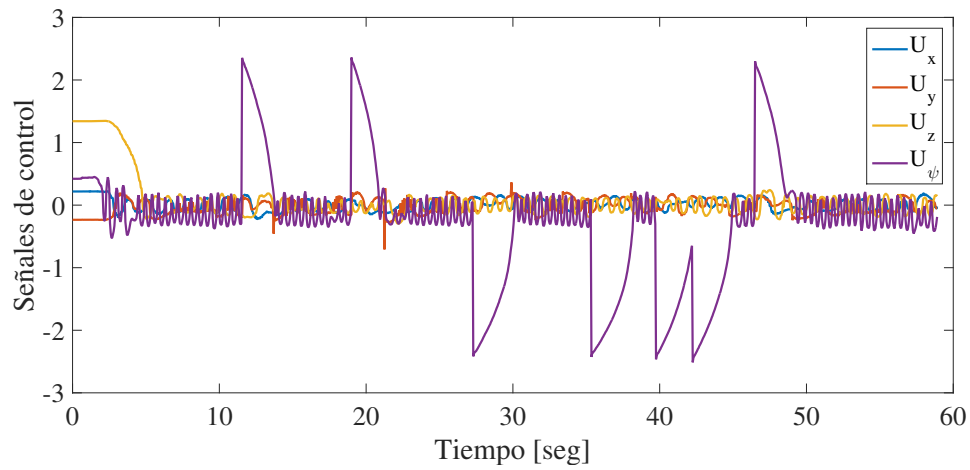


Figura 6.47: Señales de control para el Agente 2.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: <https://youtu.be/7CHxNxESAFk>

6.3 Propuesta de *Consensus no-lineal* en ley de control *Super Twisting*

En esta sección se presentan una serie de experimentos utilizando el esquema de coordinación presentado en la sección 4.4.2. A diferencia de lo visto en el *Consensus de segundo orden* (ver 6.2), ahora se realiza la coordinación de los agentes de manera interna en el controlador *Super Twisting* modificado (4.13) que utiliza la superficie deslizante propuesta (4.12). De esta manera, ya no se tiene un gestor que realice el procesamiento de la información y luego la distribuya, en su lugar, los controladores intercambian información local con los controladores vecinos y ajustan internamente la superficie deslizante para llegar al *consensus*. Lo anterior es visto de manera gráfica en la Figura 6.48.

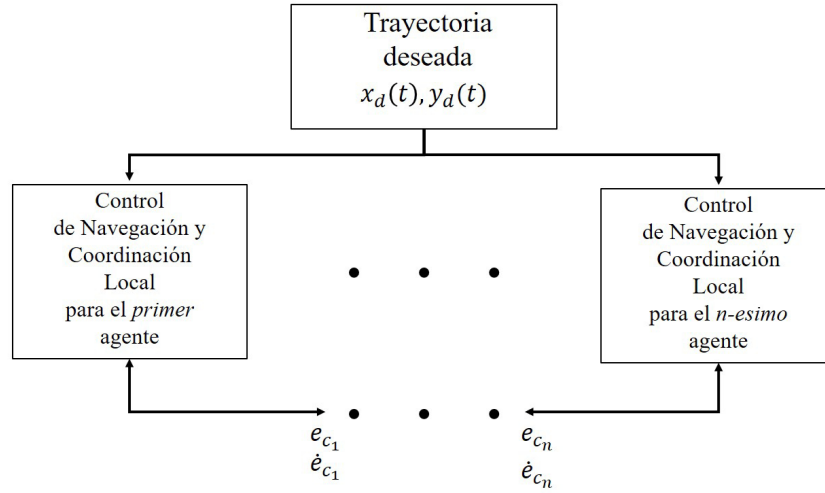


Figura 6.48: Arquitectura del sistema de coordinación utilizando solo controladores locales.

Para este caso, el grafo de conectividad utilizado en la realización de los experimentos es el mismo mostrado en la Figura 6.20, por lo que las matrices de *adyacencia* y *laplaciana* se mantienen iguales para dos agentes, y para el caso de tres, se presentan

a continuación:

$$A = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad L = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}$$

6.3.1 Trayectoria circular con dos agentes (con perturbación)

Al igual que en la sección 6.2.1, se realiza una trayectoria circular con dos cuadrirrotores, y utilizando las ecuaciones paramétricas de una circunferencia, se obtiene una trayectoria deseada cuyos datos son iguales a los presentados en dicha sección.

Igualmente, los cuadrirrotores son perturbados cada uno en instantes distintos para visualizar el efecto de la coordinación teniendo como resultado las trayectorias mostradas en la Figura 6.49.

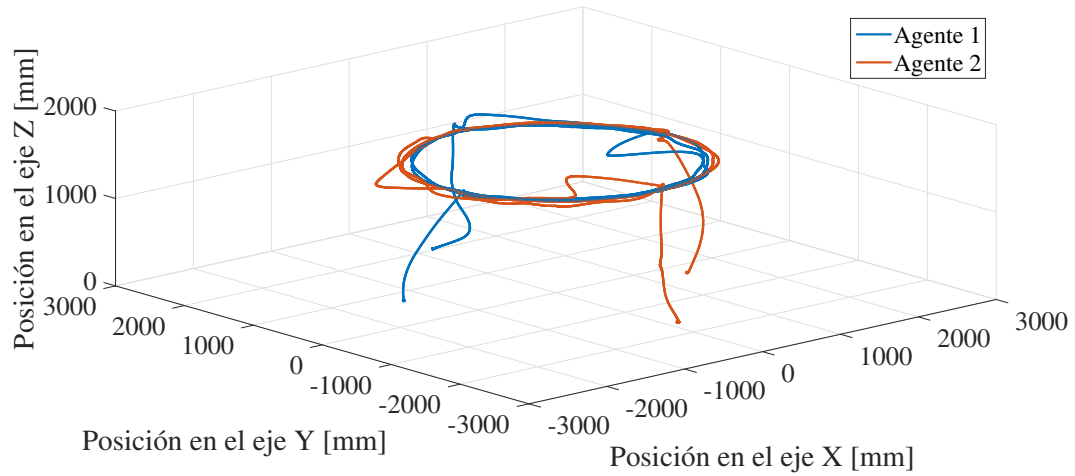


Figura 6.49: Trayectoria circular con *Consensus no-lineal* en el espacio tridimensional.

Los ángulos de *guiñada* de los agentes son tangentes a la trayectoria circular y son mostrados en la Figura 6.50 para el Agente 1 y en la Figura 6.51 para el Agente 2.

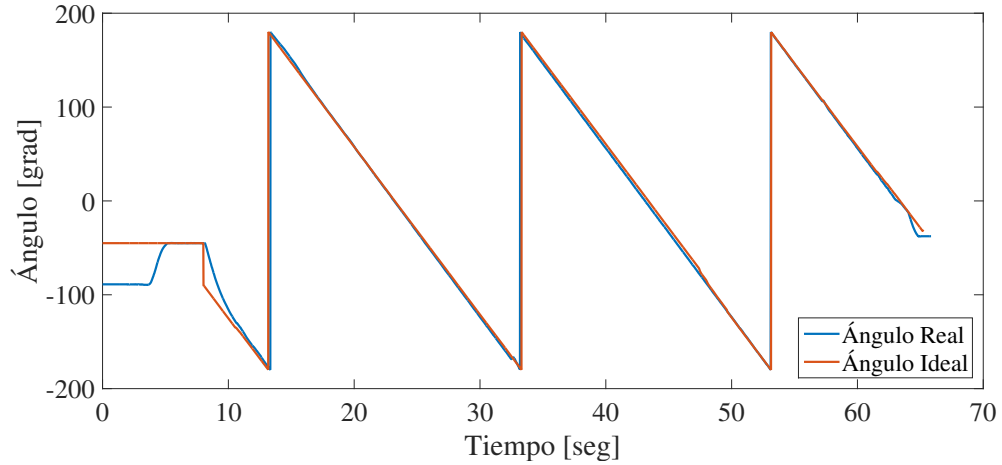


Figura 6.50: Ángulo de guiñada para el Agente 1.

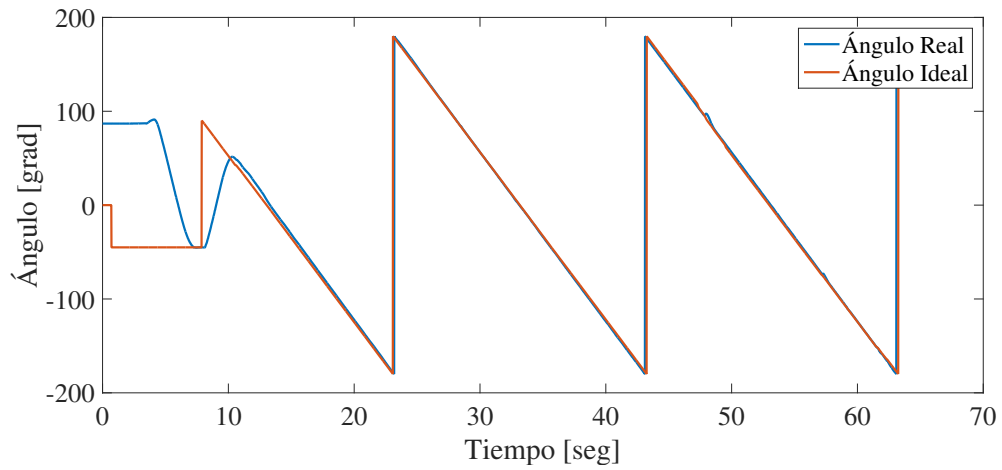
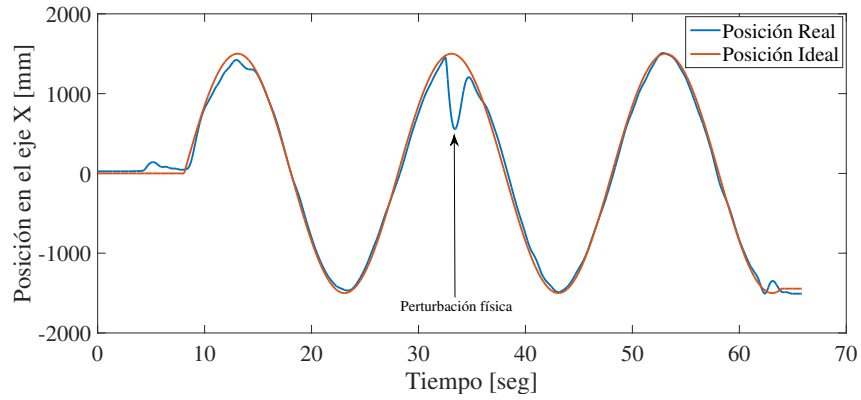


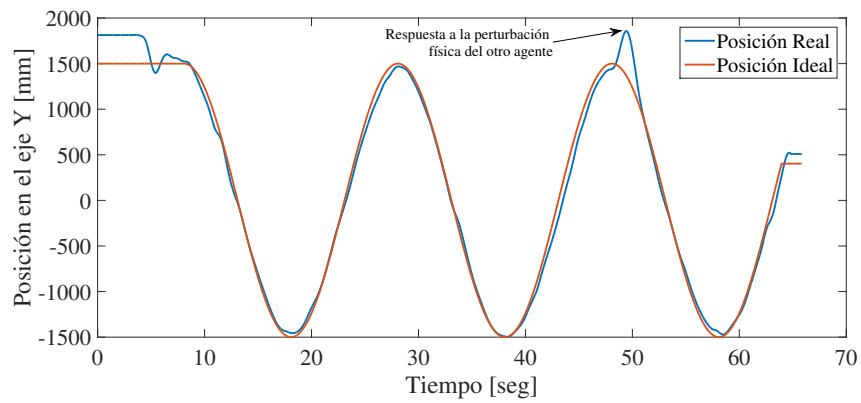
Figura 6.51: Ángulo de guiñada para el Agente 2.

Posteriormente, se muestran las posiciones en los ejes coordenados para el Agente 1 en la Figura 6.52, mientras que para el Agente 2 estas corresponden a la Figura 6.53.

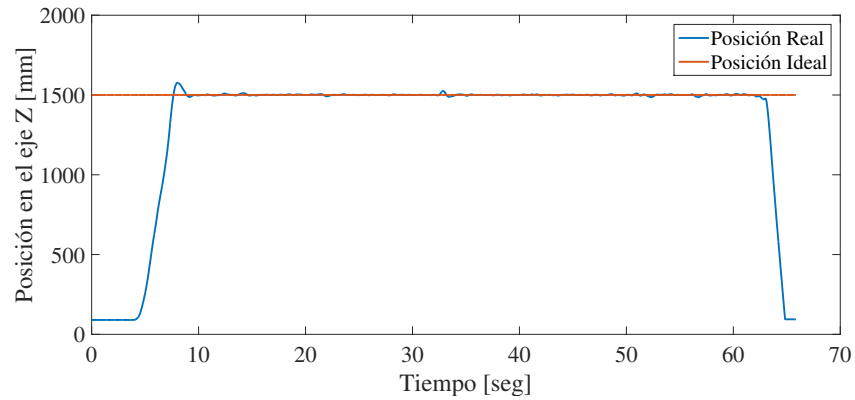
Es posible notar en las distintas figuras correspondientes a los ejes coordenados X y Y el como afectan las perturbaciones hechas sobre un agente al otro; esto es, que el agente que no fue perturbado sigue al otro agente y busca igualar su posición y velocidad.



(a) Posición en el eje X.

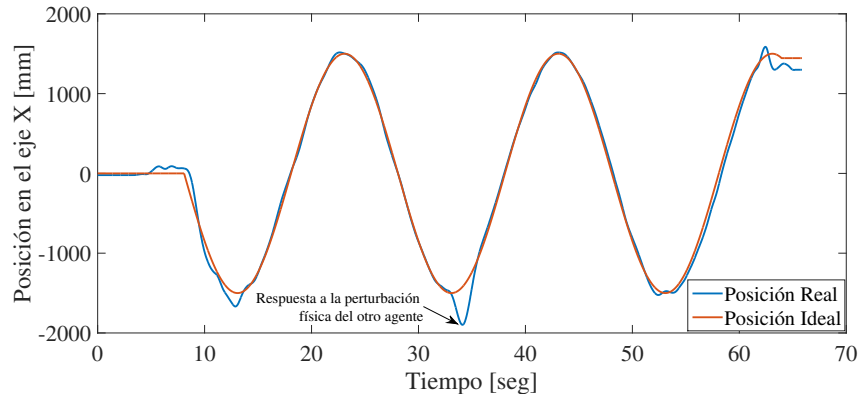


(b) Posición en el eje Y.

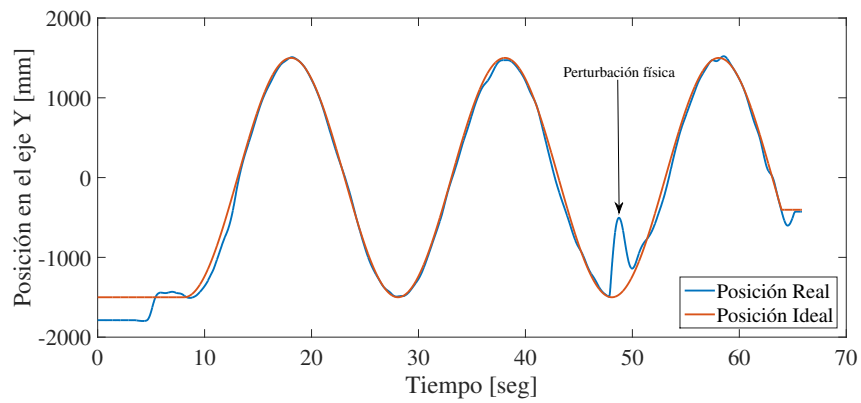


(c) Posición en el eje Z.

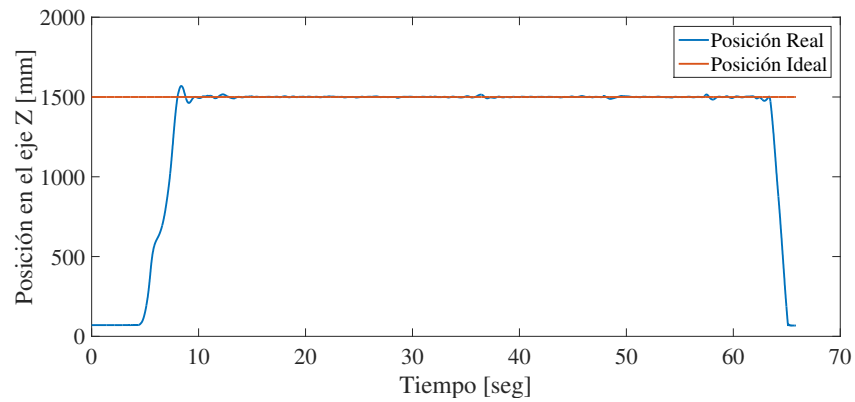
Figura 6.52: Gráficas de posiciones XYZ por eje para la trayectoria circular con *Consensus no-lineal* del Agente 1.



(a) Posición en el eje X.



(b) Posición en el eje Y.



(c) Posición en el eje Z.

Figura 6.53: Gráficas de posiciones XYZ por eje para la trayectoria circular con *Consensus no-lineal* del Agente 2.

Las superficies deslizantes producidas para los respectivos controladores se muestran

en las Figuras 6.54 y 6.55, para el Agente 1 y Agente 2, respectivamente.

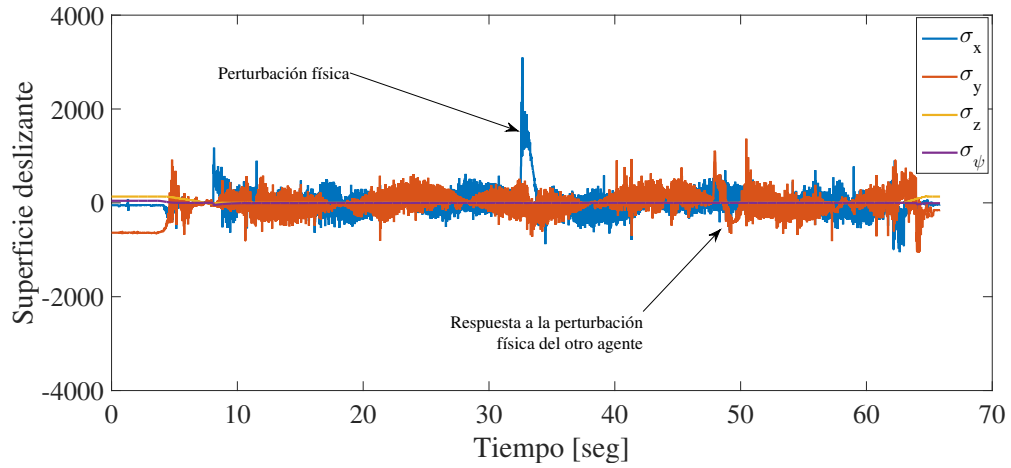


Figura 6.54: Superficies deslizantes para el Agente 1.

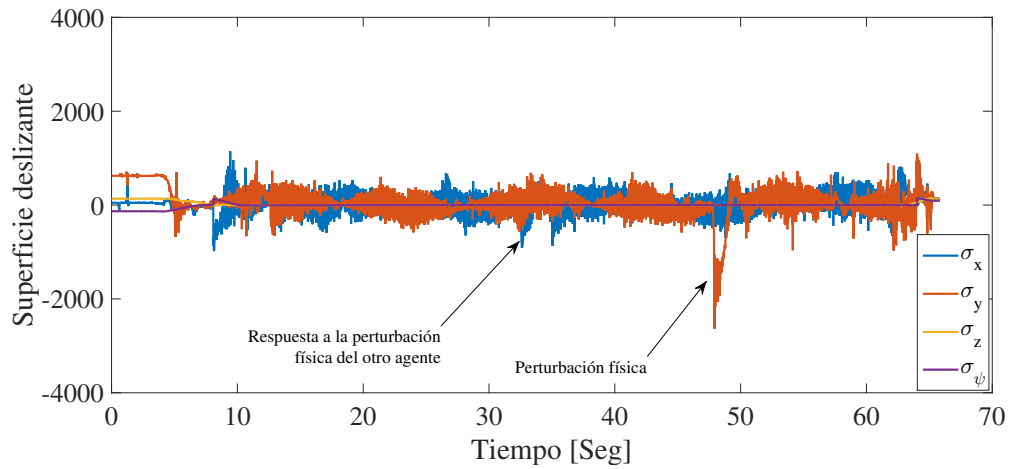


Figura 6.55: Superficies deslizantes para el Agente 2.

Usando estas superficies deslizantes, se generan las señales de control correspondientes para cada agente como se muestran en las Figuras 6.56 y 6.57.

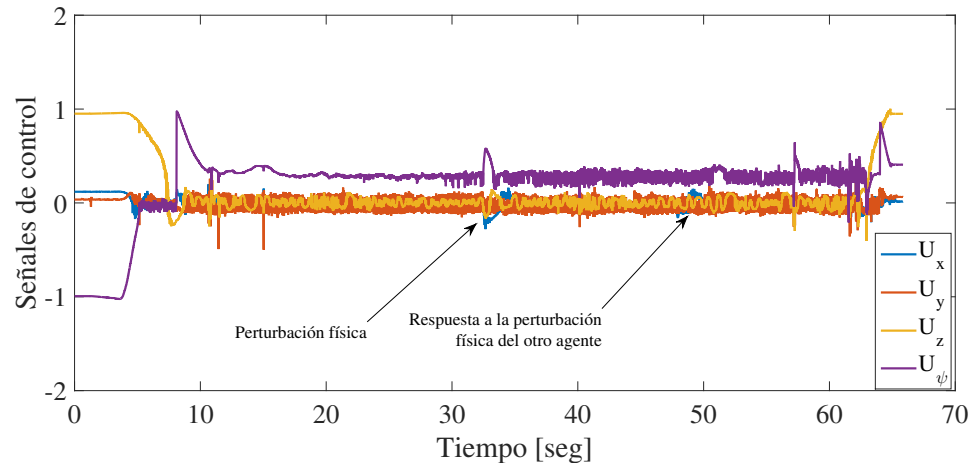


Figura 6.56: Señales de control para el Agente 1.

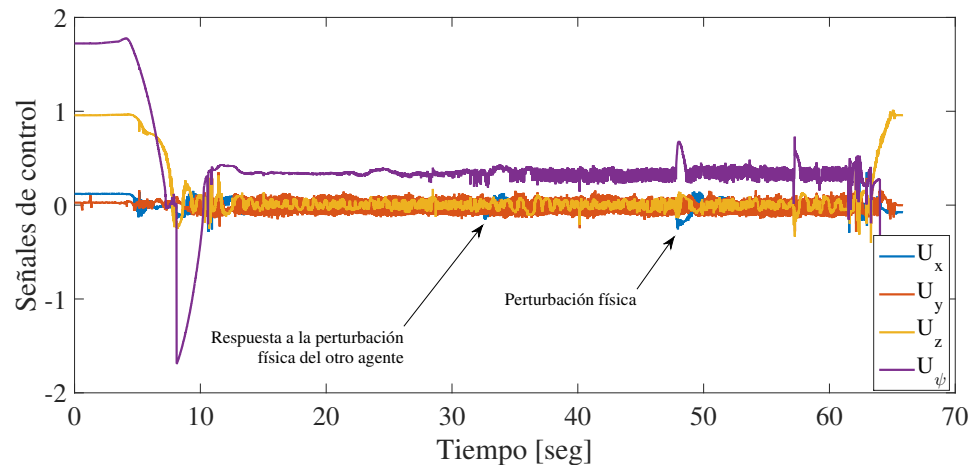


Figura 6.57: Señales de control para el Agente 2.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: <https://youtu.be/cQ6UMtUKdDM>

6.3.2 Trayectoria para cubrir un área cuadrada

Este experimento consiste en la realización de una trayectoria similar a la vista en la Figura 6.36; esto es, una trayectoria para cubrir un área cuadrada utilizando dos cuadrirrotores para realizar una tarea en un menor número de vueltas.

A diferencia del experimento de la sección 6.2.2, el valor de la altura de referencia es ahora de 1500 mm y se utiliza el algoritmo de *Consensus no-lineal* para realizar la coordinación de velocidad y posición de los agentes.

Los resultados experimentales en el espacio tridimensional son mostrados en la Figura 6.58, donde se puede observar el seguimiento de la trayectoria por parte de los dos agentes.

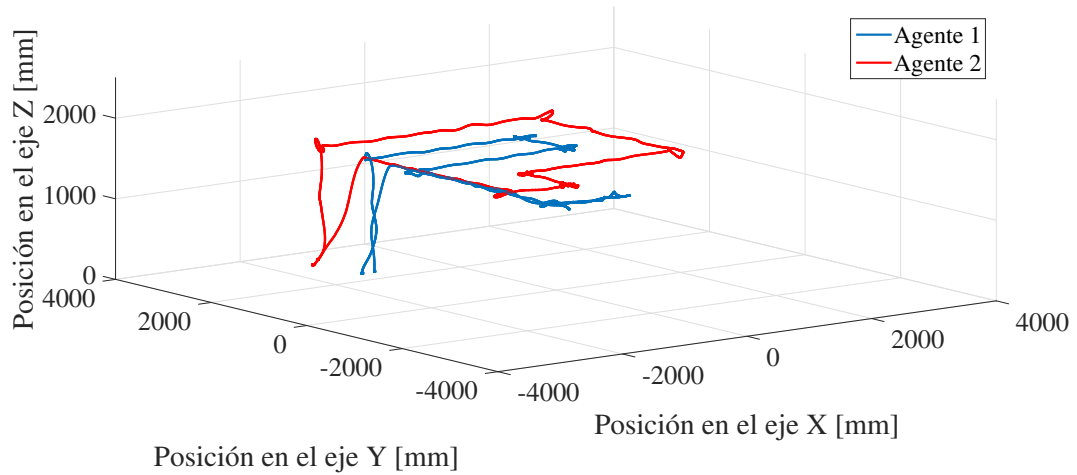


Figura 6.58: Seguimiento real para cubrir un área cuadrada.

Los ángulos de guiñada de los agentes son tangentes a la trayectoria y cambian cuando llegan a sus respectivos puntos de giro. Los valores de estos ángulos pueden ser vistos en las Figuras 6.59 y 6.60.

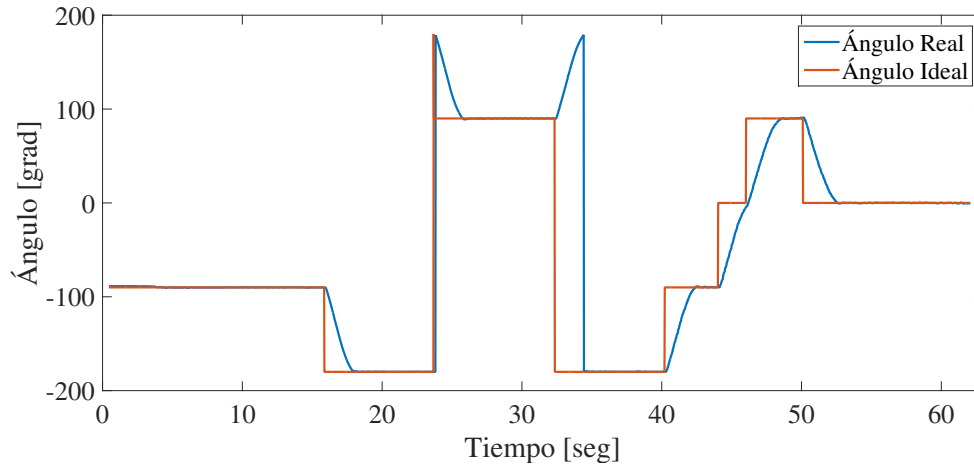


Figura 6.59: Ángulo de guiñada para el Agente 1.

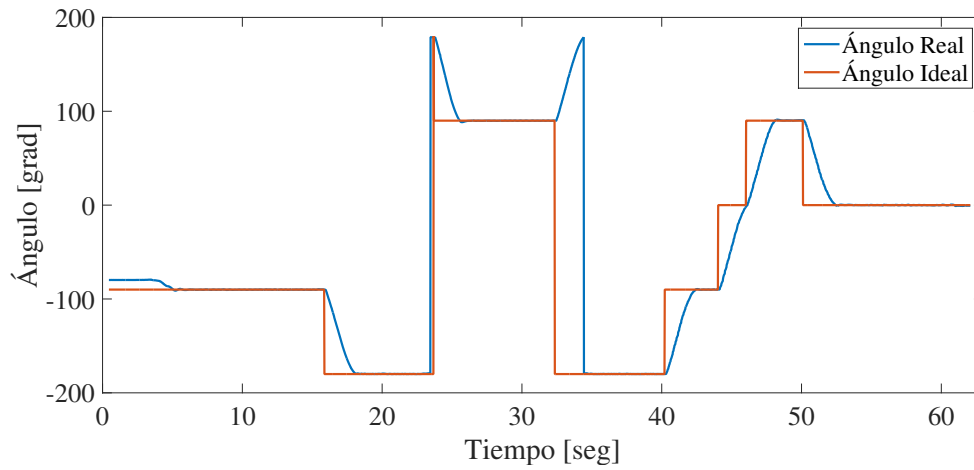
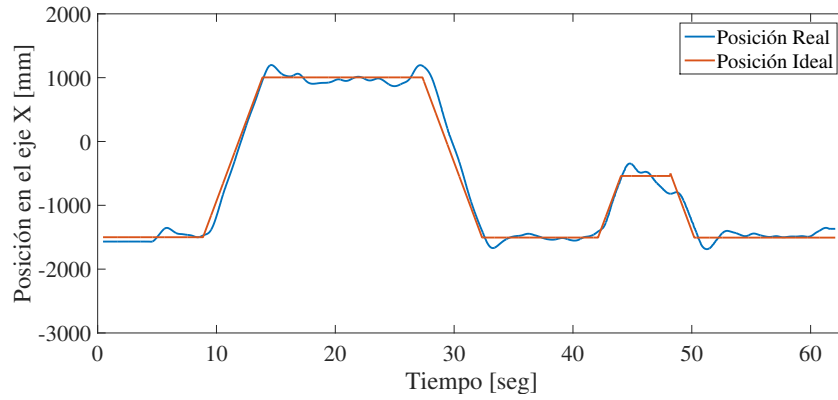
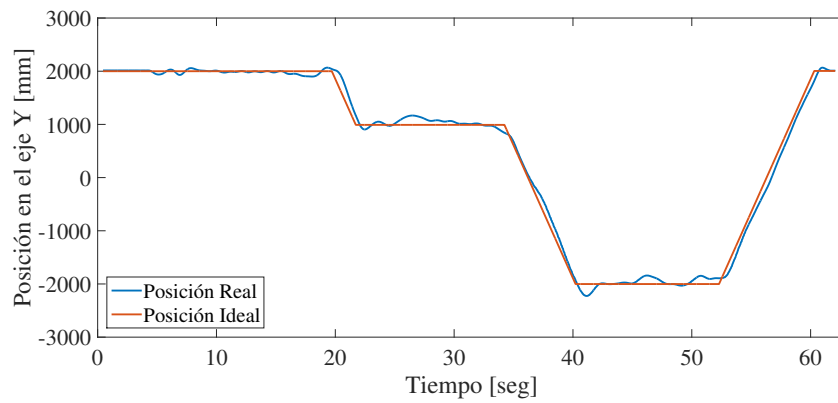


Figura 6.60: Ángulo de guiñada para el Agente 1.

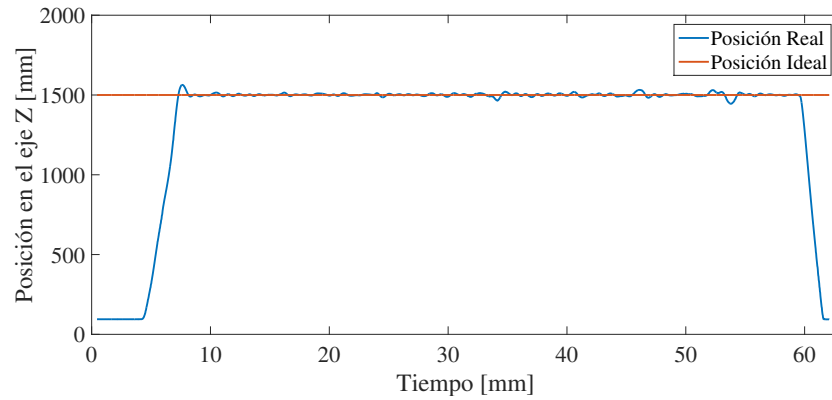
En la Figura 6.61 es posible observar el comportamiento del Agente 1 en cada uno de los ejes coordenados, y la Figura 6.62 corresponde a los datos para el Agente 2.



(a) Posición en el eje X.

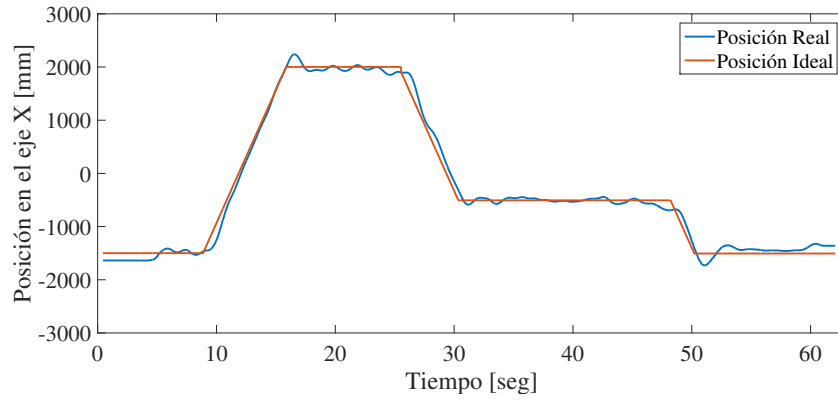


(b) Posición en el eje Y.

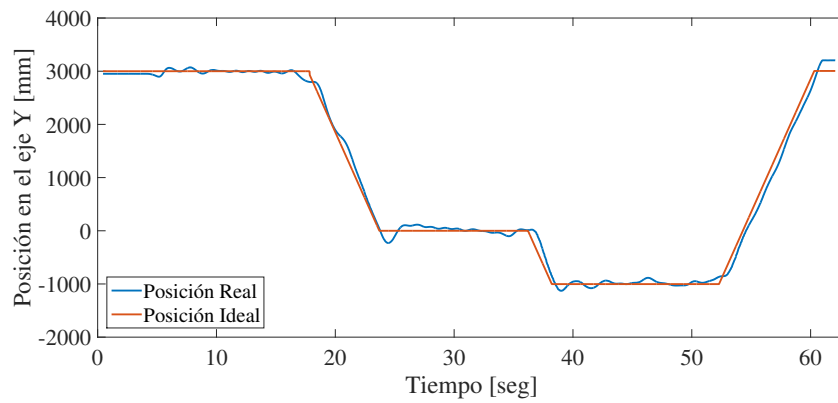


(c) Posición en el eje Z.

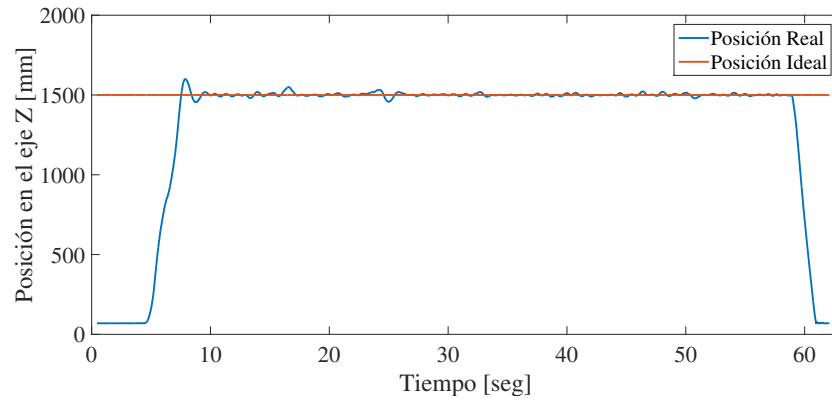
Figura 6.61: Gráficas de posiciones XYZ por eje para la trayectoria para cubrir un área cuadrada con *Consensus no-lineal* del Agente 1.



(a) Posición en el eje X.



(b) Posición en el eje Y.



(c) Posición en el eje Z.

Figura 6.62: Gráficas de posiciones XYZ por eje para la trayectoria para cubrir un área cuadrada con *Consensus no-lineal* del Agente 2.

Las superficies deslizantes generadas debido al seguimiento de esta trayectoria son

mostradas en la Figura 6.63 para el Agente 1 y en la Figura 6.64 para el Agente 2.

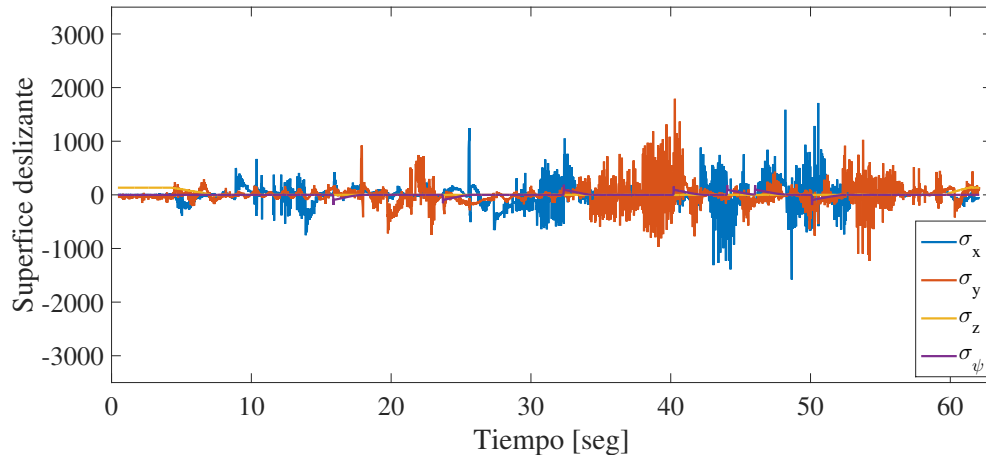


Figura 6.63: Superficies deslizantes para el Agente 1.

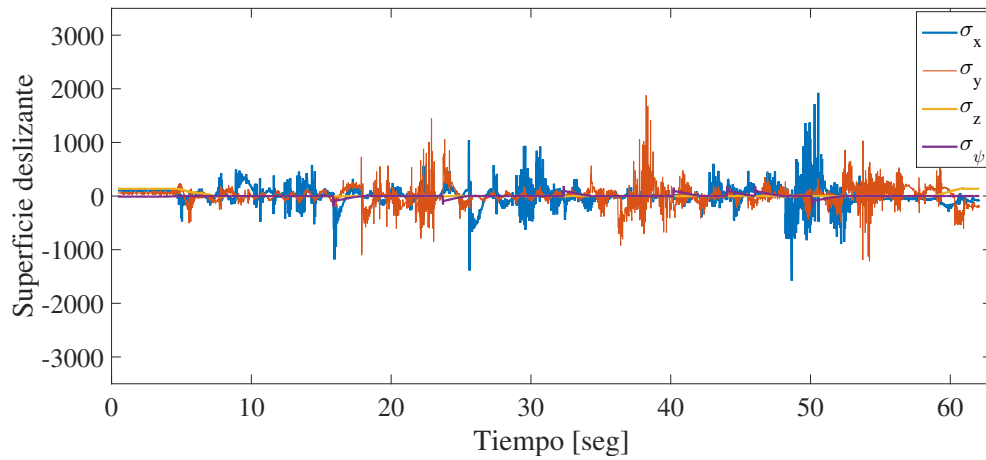


Figura 6.64: Superficies deslizantes para el Agente 2.

Haciendo uso de las superficies deslizantes mostradas, se generan los controles correspondientes para cada agente, los cuales son mostrados en las Figuras 6.65 y 6.66, para el Agente 1 y el Agente 2, respectivamente.

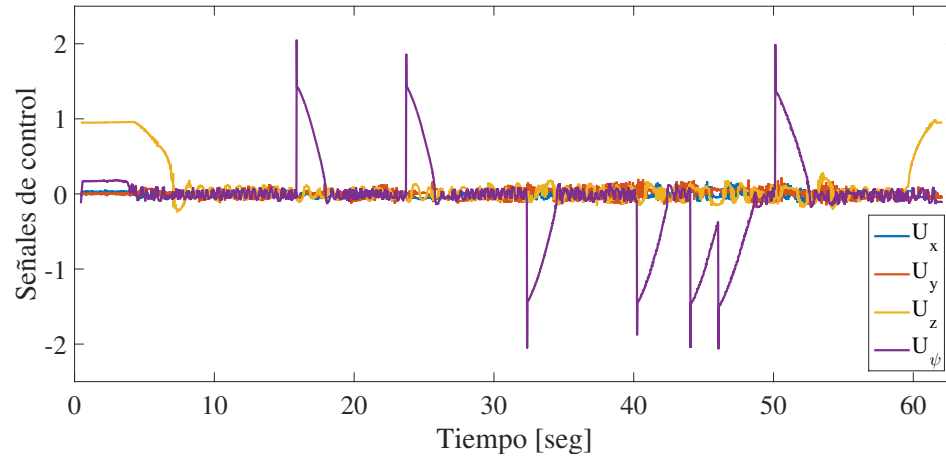


Figura 6.65: Señales de control para el Agente 1.

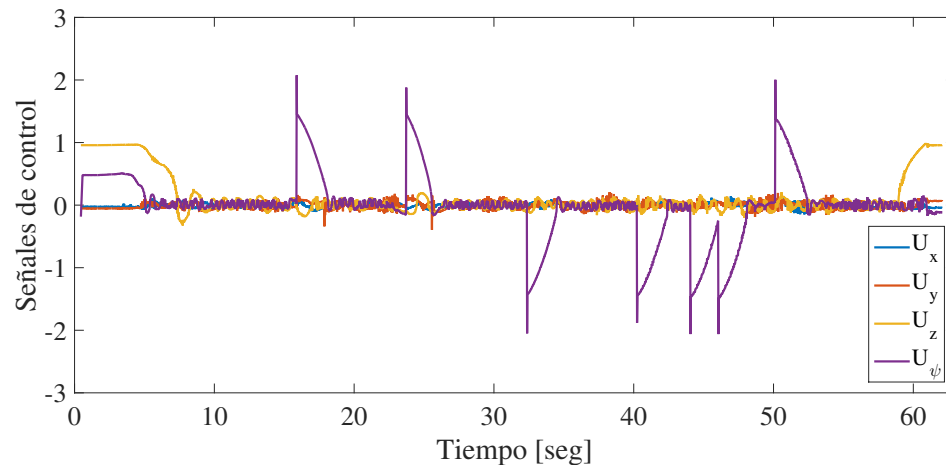


Figura 6.66: Señales de control para el Agente 2.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: https://youtu.be/01M_fdWLpEo

6.3.3 Trayectoria en forma de lemniscata con tres agentes en persecución

En este experimento se realiza una formación en tipo persecución; esto es, se tiene un agente que va al frente de los demás mientras que los que lo siguen describen la misma trayectoria pero con una separación establecida, y realizando las mismas maniobras y movimientos que el primer agente realiza.

Para el caso de la trayectoria que sigue esta formación, se opta por una lemniscata, ver Figura 6.67, debido a que es una trayectoria cíclica y continuamente diferenciable.

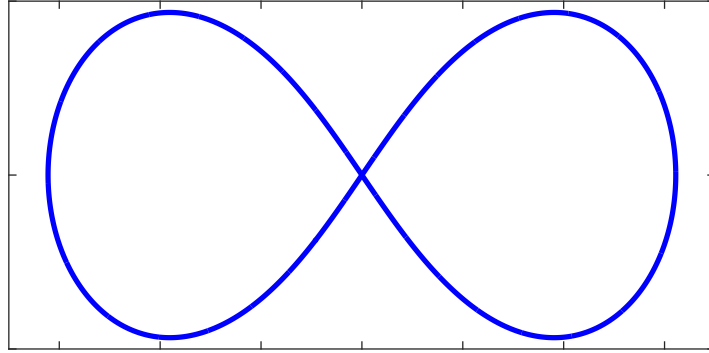


Figura 6.67: Lemniscata.

Las ecuaciones paramétricas de la lemniscata utilizadas para la generación de las coordenadas en el plano XY , presentadas en [27], se muestran a continuación:

$$x_{lm} = \frac{a\sqrt{2}\cos(2\pi f_{lm}t)}{\sin^2(2\pi f_{lm}t) + 1}; \quad y_{lm} = \frac{a\sqrt{2}\cos(2\pi f_{lm}t)\sin(2\pi f_{lm}t)}{\sin^2(2\pi f_{lm}t) + 1} \quad (6.8)$$

donde x_{lm} y y_{lm} son las coordenadas resultantes de la trayectoria, t es el tiempo, a corresponde a la distancia focal y f_{lm} corresponde al número de veces que se realiza la trayectoria por segundo. En el caso de este experimento, los siguientes datos son utilizados:

- Distancia focal de 2200 *mm* desde el origen.
- Coordenadas del centro de la lemnistaca en el origen.
- Velocidad de 0.04 *vuelatas/seg* (2.4 *vuelatas/min*).

Tomando las ecuaciones 6.8, utilizando los datos ya vistos, como referencias para el algoritmo de *consensus no-lineal* se obtienen los datos mostrados en la Figura 6.68 para una persecución de 3 agentes.

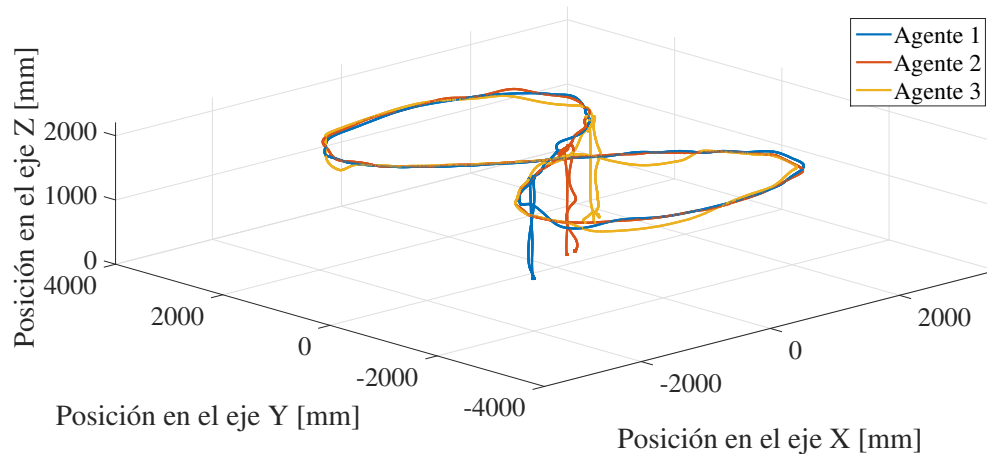


Figura 6.68: Trayectoria descrita por los tres agentes realizando una lemniscata.

Los ángulos de *guiñada* de cada agente son tangentes a la trayectoria y pueden ser vistos en las Figuras 6.69, 6.70 y 6.71 para el Agente 1, Agente 2, y Agente 3, respectivamente.

Es posible notar que debido a que los agentes recorren la misma trayectoria pero con un desfase de tiempo, los ángulos de guiñada son iguales para los tres agentes solo que recorridos en el tiempo.

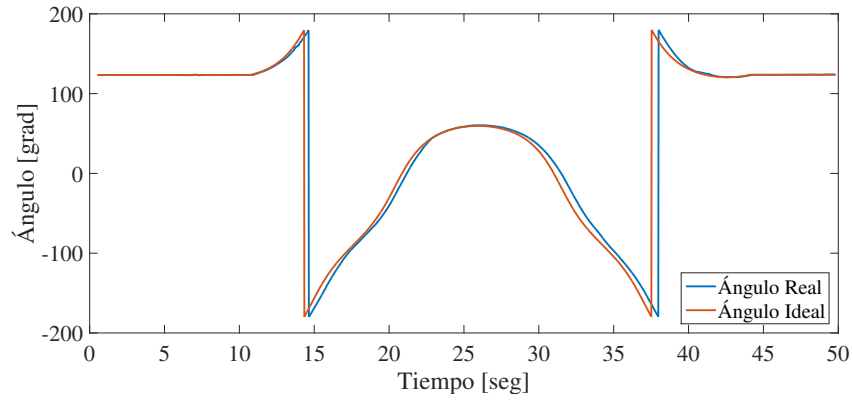


Figura 6.69: Ángulo de guiñada del Agente 1.

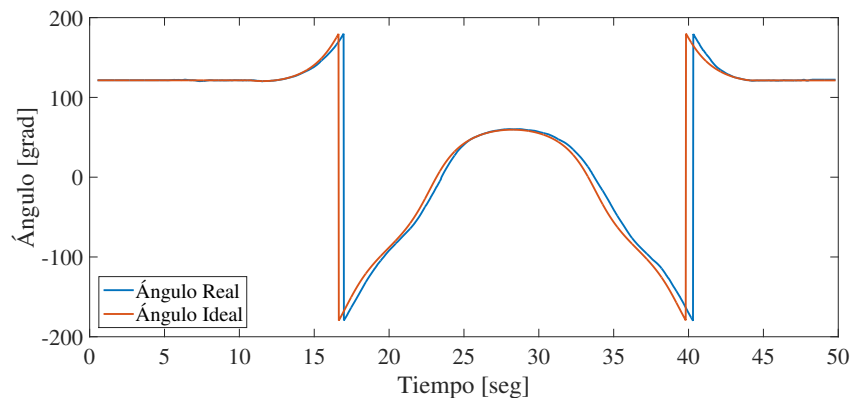


Figura 6.70: Ángulo de guiñada del Agente 2.

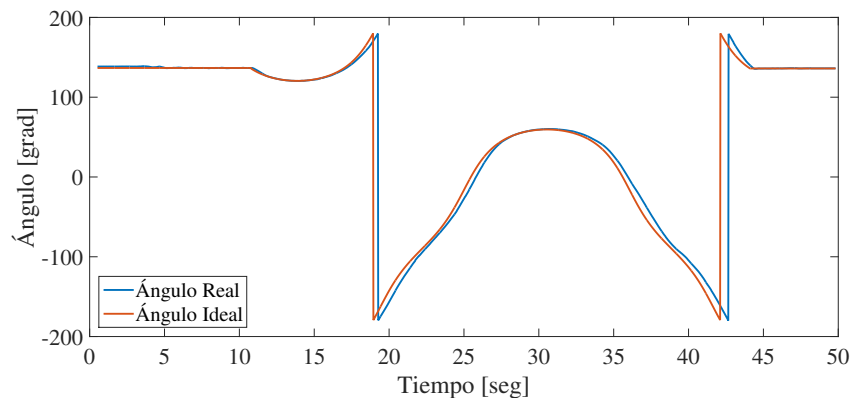
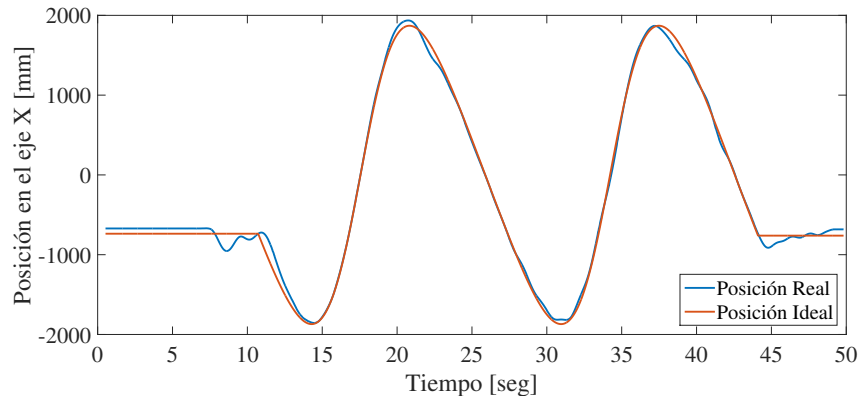
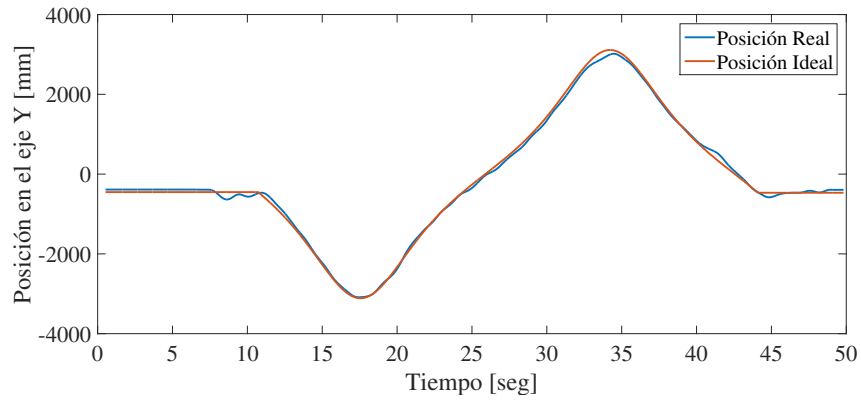


Figura 6.71: Ángulo de guiñada del Agente 3

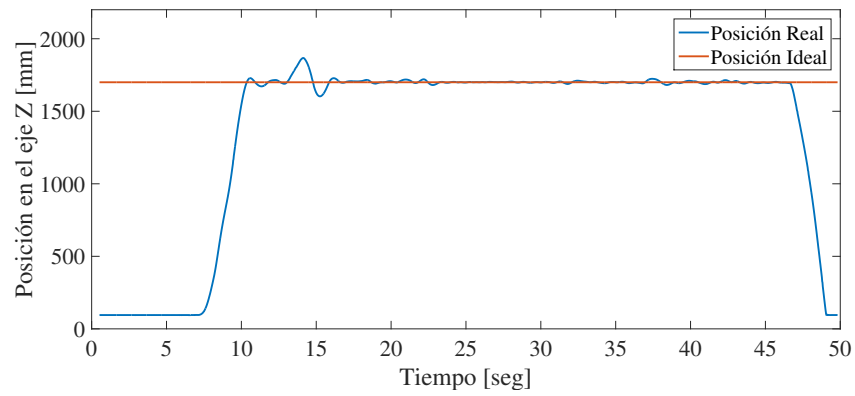
En la Figura 6.72 se muestra la trayectoria para el Agente 1 en cada uno de sus ejes coordenados.



(a) Posición en el eje X.



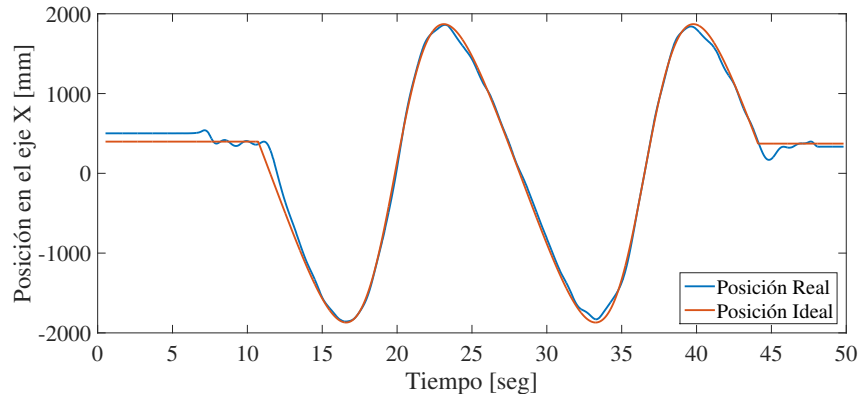
(b) Posición en el eje Y.



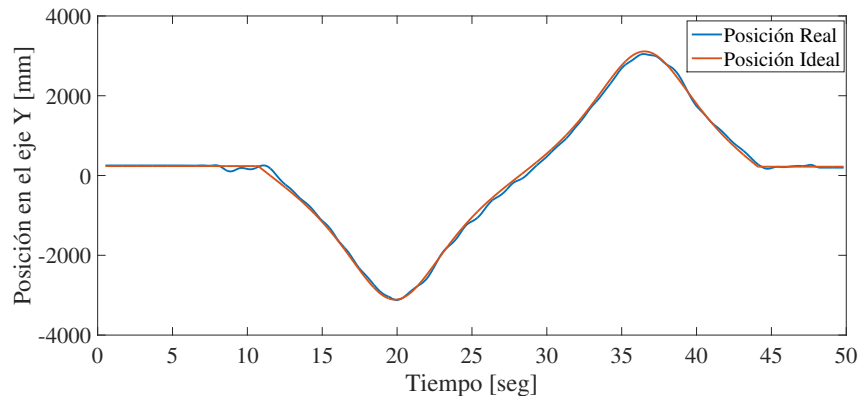
(c) Posición en el eje Z.

Figura 6.72: Gráficas de posiciones XYZ por eje para la trayectoria en forma de *Lemniscata* con *Consensus no-lineal* del Agente 1.

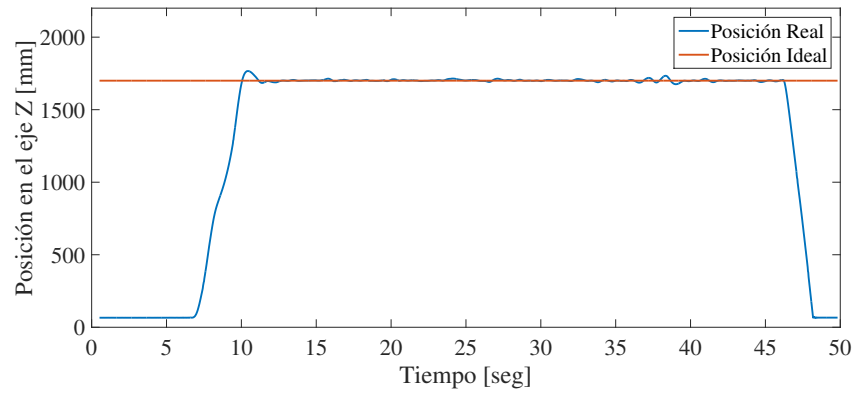
Para el Agente 2, las posiciones en los ejes X , Y y Z se muestran en la Figura 6.73.



(a) Posición en el eje X.



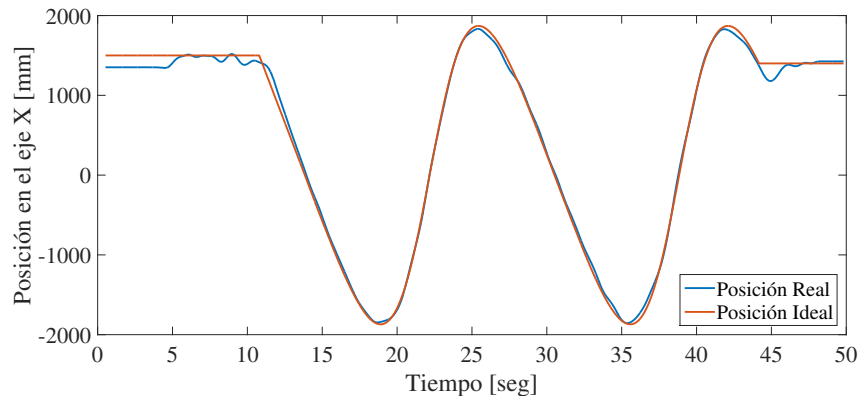
(b) Posición en el eje Y.



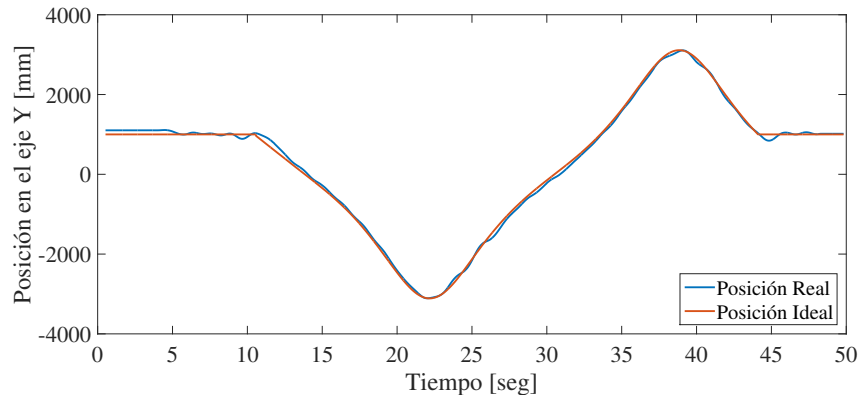
(c) Posición en el eje Z.

Figura 6.73: Gráficas de posiciones XYZ por eje para la trayectoria en forma de *Lemniscata* con *Consensus no-lineal* del Agente 2.

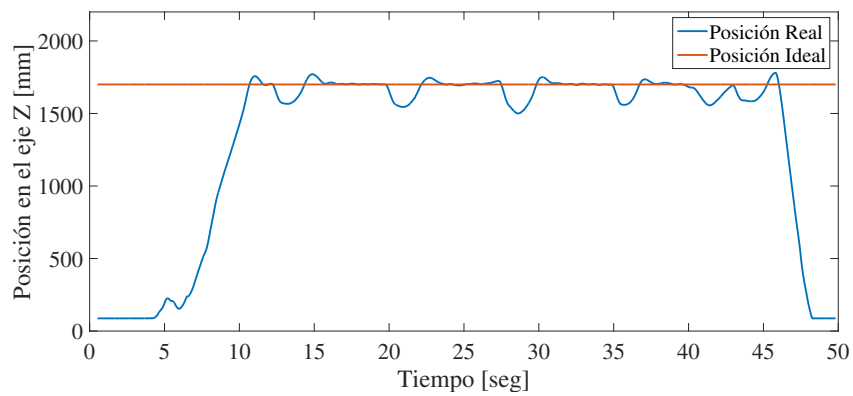
Finalmente, para el Agente 3 se muestran las posiciones en los ejes coordenados en la Figura 6.74.



(a) Posición en el eje X.



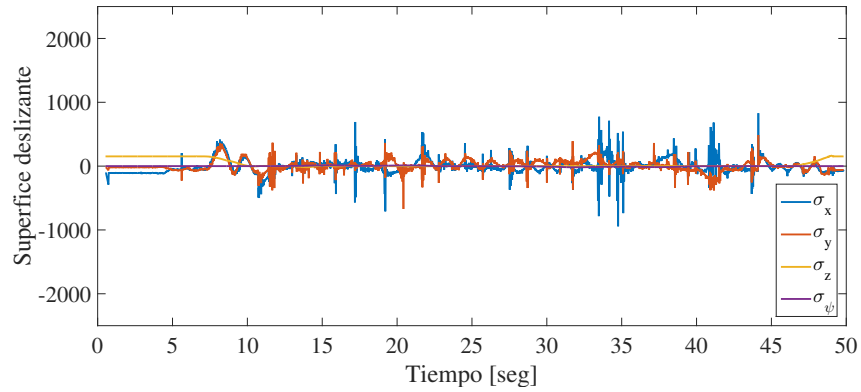
(b) Posición en el eje Y.



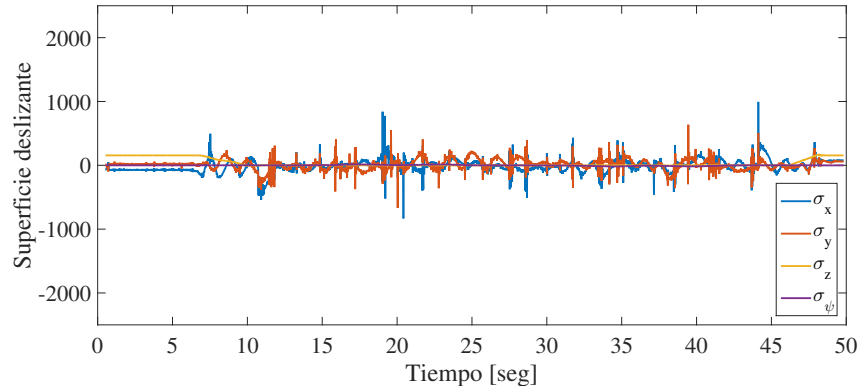
(c) Posición en el eje Z.

Figura 6.74: Gráficas de posiciones XYZ por eje para la trayectoria en forma de *Lemniscata* con *Consensus no-lineal* del Agente 3.

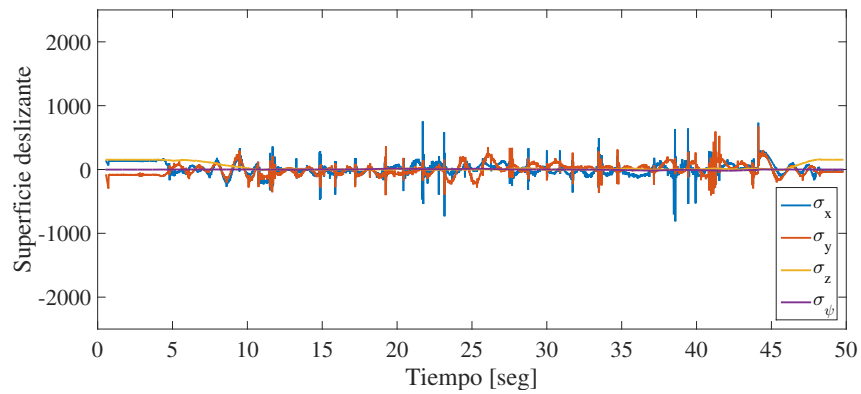
Las superficies deslizantes utilizadas en los controladores de cada uno de los agentes se muestran en la Figura 6.75.



(a) Superficies deslizantes para el Agente 1.



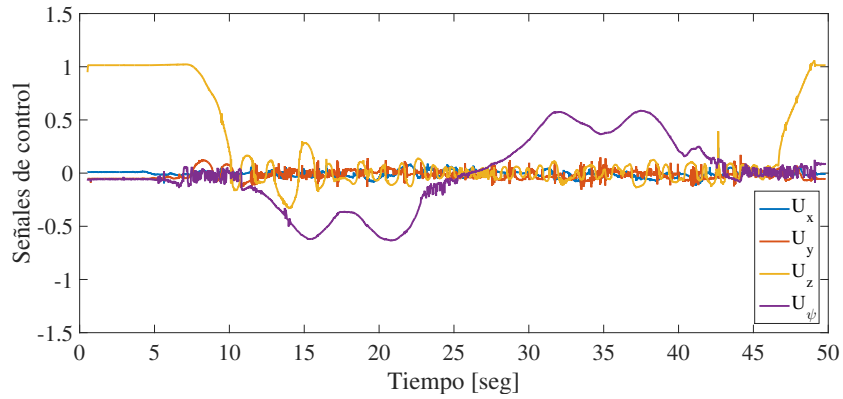
(b) Superficies deslizantes para el Agente 2.



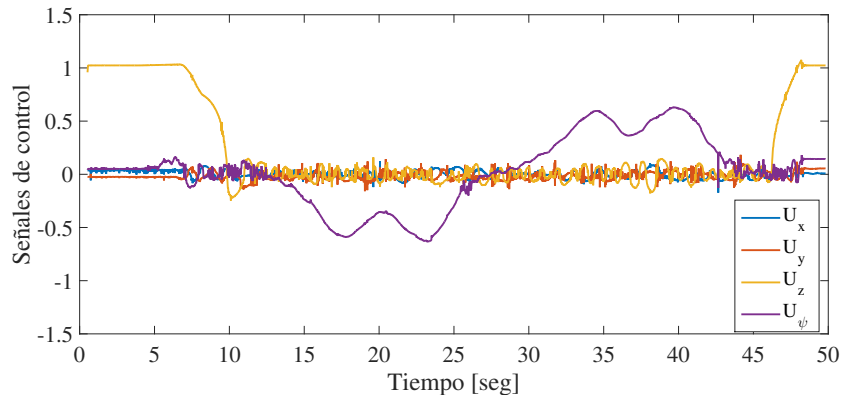
(c) Superficies deslizantes para el Agente 3.

Figura 6.75: Superficies deslizantes para los 3 agentes realizando una *Lemniscata*.

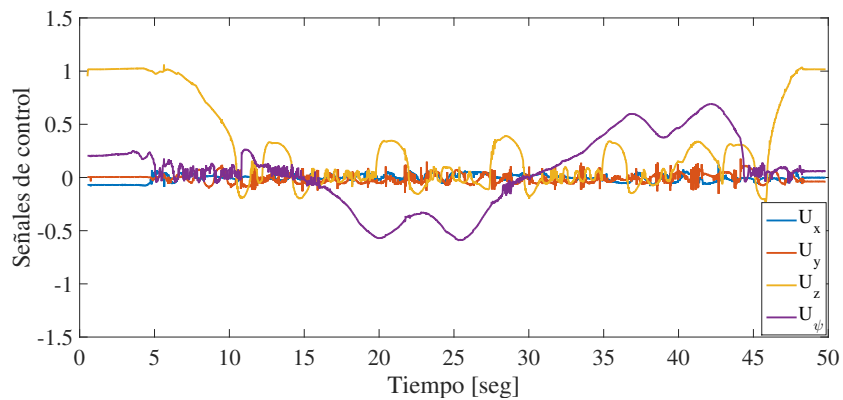
Utilizando las superficies deslizantes, se generan las señales de control para los tres agentes las cuales son mostradas en la Figura 6.76.



(a) Señales de control para el Agente 1.



(b) Señales de control para el Agente 2.



(c) Señales de control para el Agente 3.

Figura 6.76: Señales de control para los 3 agentes realizando una *Lemniscata*.

Este experimento fue capturado en video y puede ser visto en la siguiente dirección *Web*.

Hipervínculo a *YouTube*: <https://youtu.be/Ijqo0qU25vY>

Capítulo 7

Sumario, Conclusiones y Recomendaciones

7.1 Sumario

A lo largo de este trabajo de investigación se abordaron los distintos tópicos necesarios para llegar al objetivo propuesto; esto es, el vuelo en formación de multi-agentes.

Como paso inicial, se describieron las generalidades de los *Vehículos Aéreos No Tripulados* para tener una correcta comprensión de sus características, ventajas, y de igual manera, inconvenientes que se pudieran presentar con su utilización.

Además, se analizaron las ecuaciones de movimiento correspondientes a un vehículo tipo *cuadrirrotor*, describiendo detalladamente los elementos que las conforman, para de esta manera desarrollar los algoritmos de *Guiado, Control y Navegación*.

En capítulos posteriores, se desarrollaron las estrategias de *Guiado, Navegación y Control*, utilizando las ecuaciones de movimiento previamente analizadas, así como los conceptos básicos del *control por modos deslizantes* y su variante *Super Twisting*, presentados en los Apéndices.

Una vez expuesto lo referente al vehículo utilizado, se introdujo a los conceptos del vuelo en formación, así como a las herramientas necesarias para lograr la coordina-

ción del sistema de multi-agentes. En esta misma sección se presentan los algoritmos de coordinación utilizados comúnmente, y se propone el algoritmo de *Consensus no-lineal* por medio de control *Super Twisting*.

Finalmente, se describe de manera breve la plataforma experimental utilizada, sus componentes y arquitectura básica para después realizar una serie de experimentos utilizando los algoritmos propuestos.

7.2 Conclusiones y discusión de resultados

Durante los experimentos realizados en la sección 6.1 se pudo observar el comportamiento y desempeño adecuados de las estrategias de *Guiado, Navegación y Control*, lo que denota su correcta aplicación e implementación en el sistema físico. Además, el controlador *Super Twisting* demostró tener rapidez de respuesta y robustez ante perturbaciones externas.

De igual forma, la utilización del algoritmo de *consensus* de segundo orden, visto en la sección 6.2, resultó tener un desempeño aceptable para la coordinación de los agentes, comprobando que es posible llegar al *consensus* a nivel de referencias de posición; sin embargo, el uso de esta estrategia no garantiza un comportamiento destacable frente a las perturbaciones rápidas y súbitas. Además, el hecho de tener oscilaciones de alta frecuencia en la referencia de posición generada por el *Gestor consensus*, puede inestabilizar a los controladores locales dadas las altas velocidades de oscilación, por lo que solo es recomendable su utilización cuando no se está en presencia de perturbaciones de gran magnitud.

En la sección 6.3 se realizaron experimentos para probar el desempeño de la propuesta de *Consensus no-lineal* basado en el control *Super Twisting*. En este algoritmo el comportamiento ante perturbaciones externas fue más adecuado que con el primer algoritmo de coordinación implementado, dado que presentó mayor rapidez de respuesta y un seguimiento de posición de mayor magnitud, lo que significa que la

formación mantiene su geometría en caso de perturbación de alguno de los agentes. También se pudo notar que el seguimiento de la trayectoria fue más suave y se presencia menos error en las posiciones resultantes. Además, debido a que la coordinación se realiza a nivel de control local, y la referencia es directa de la trayectoria, no se presentaron las desventajas vistas con el uso de un *Gestor de coordinación*, y además, en caso de fallo, la formación puede continuar debido a la ejecución local de la coordinación; no obstante, también se pudieron observar algunas desventajas con la utilización de este esquema, como lo fueron los cambios rápidos en las *superficies deslizantes* y por consiguiente en las señales de control, así como la alta sensibilidad frente a pequeños cambios o perturbaciones en la formación, que se traducen en «ruido» en las señales de control.

7.3 Recomendaciones para trabajo futuro

En este trabajo de investigación se realizó la implementación de algoritmos de *Consensus* para la coordinación de un sistema de multi-agentes. A pesar de esto, todavía es posible realizar mejoras en los algoritmos para garantizar su buen desempeño bajo circunstancias no contempladas en los alcances de este trabajo, así como la adición de más características al sistema de multi-agentes. Estas recomendaciones y agregables son presentados a continuación:

- Es necesario considerar las perturbaciones físicas para el sistema como un elemento en las ecuaciones de *Consensus* para responder de una manera más predecible ante las mismas. Lo anterior puede ser realizado mediante la caracterización de las perturbaciones comunes presentes en el tipo de vehículo utilizado, que para el caso de cuadrirrotores, es posible considerar perturbaciones debido al *efecto tierra*, *viento cruzado*, y rafagas de viento producidas por los mismos agentes de la formación.
- Para solucionar el problema del «ruido» en las señales de control utilizando

el algoritmo de *Consensus no-lineal*, es posible implementar varios métodos para ignorar los pequeños cambios y perturbaciones que pueden ser despreciables, como lo son el uso de funciones de *saturación*, filtrado de la información intercambiada entre los agentes, así como adaptabilidad en las ganancias del algoritmo.

- Finalmente es recomendable la implementación de estos algoritmos en formaciones y trayectorias más complejas, contemplando el vuelo en ambientes exteriores para evaluar el comportamiento frente a condiciones de operación más realistas.

Apéndice A

Control por modos deslizantes

Como se ha mencionado en la sección 2.1, el modelo matemático de un VANT, de tipo cuadrirrotor, es *No-lineal* y *subactuado*¹. Es por esto que se vuelve necesario la utilización de un controlador que garantice la robustez del sistema ante todas las condiciones modeladas, así como las perturbaciones debido al viento.

El control por *modos deslizantes* es una técnica de control que como principales ventajas tiene robustez del sistema, convergencia en tiempo finito y dinámicas de compensación de orden reducido [28]. Además, es una técnica de control ampliamente utilizada en distintas aplicaciones, debido a su versatilidad y fácil adaptación a diversos sistemas.

Un controlador por modos deslizantes tiene como principal objetivo de diseño conducir los estados de un sistema específico sobre la *superficie deslizante* [29] la cual toma su nombre debido a que en un plano de fase describe una línea por la cual los estados del sistema se «deslizan» hasta llegar a la referencia, ver Figura A.1. Por lo tanto, para poder controlar un sistema, es necesario diseñar tal controlador en dos partes. La primera es la correspondiente a la superficie deslizante de tal manera que se satisfagan las especificaciones previstas en el diseño. La segunda parte corresponde a la ley de control que realizará la conmutación del sistema para permanecer en la superficie deslizante [30].

¹Cuando el número de grados de libertad es mayor a las entradas de control.

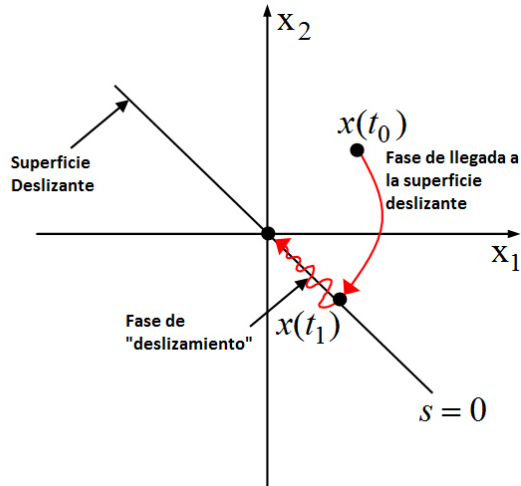


Figura A.1: Superficie deslizante en el plano de fase.

A.1 Modos Deslizantes para Sistemas de Segundo Orden

Un sistema de segundo orden, que obedece la segunda ley de Newton, es aquel que puede ser descrito como una *masa unitaria* con movimiento uni-dimensional. Este sistema es controlado por una *fuerza* que produce una *aceleración* en la masa, provocando un cambio en la *velocidad* y posteriormente produciendo un *movimiento*. Esto resulta:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u + f(x_1, x_2, t) \end{aligned} \tag{A.1}$$

donde x_1 es la posición de la masa, x_2 la velocidad, e indirectamente, u la entrada de control. Además se toma en cuenta un término de perturbación $f(x_1, x_2, t)$ como compensación para cuestiones físicas tales como fricción, fuerzas de resistencia, entre otras.

El objetivo de un controlador es mantener la respuesta o salida del sistema tratado en

alguna referencia dada. Lo anterior se conoce como *seguimiento*, *rastreo* o *tracking*². Definiendo un error e como la diferencia que existe entre la referencia y la salida medida, esto es $e = y_{ref} - y$, donde y es la salida del sistema, el controlador busca siempre minimizar tal parámetro para que de esta manera cumpla con el objetivo de seguimiento.

Para el sistema (A.1) la salida medida es el desplazamiento x_1 , por lo que el error de seguimiento queda definido de la siguiente manera:

$$e = x_{1ref} - x_1 \quad (\text{A.2})$$

Ahora se procede a la primera parte del diseño del controlador; esto es, el diseño de una superficie deslizante tomando en cuenta que debe tender a cero, $\sigma = 0$, en tiempo finito de tal manera que la solución del sistema pueda converger a cero. Por lo general, se diseña utilizando el error de seguimiento, y sus correspondientes derivadas, tomando usualmente como criterio para el orden de la superficie deslizante, el grado relativo entre la entrada y la salida del sistema [31]. Establecido esto, para el sistema (A.1) se diseña la superficie deslizante utilizando el error definido en (A.2), así como su derivada temporal la cual es la velocidad, por lo que se obtiene una superficie deslizante de forma diferencial de la siguiente manera:

$$\sigma = k_0 e + \dot{e} \quad (\text{A.3})$$

donde k_0 es una constante positiva. Además, e y \dot{e} son el error (A.2) y su respectiva derivada.

Para la segunda parte del diseño del controlador, se busca obtener una ley de control que conduzca los estados sobre la superficie deslizante, por lo que la forma general de la función es:

$$u = H_n \text{sgn}(\sigma) \quad (\text{A.4})$$

²Término en idioma inglés más comunmente utilizado.

donde H_n es una constante positiva que asegura que el sistema se mantiene en la superficie deslizante. Se puede notar que la ley de control se adapta dependiendo de donde se encuentran los estados del sistema; si la superficie deslizante es mayor a cero, la ley de control, dada la función signo, tiende a cambiar si fuera negativa, esto es:

$$u = \begin{cases} -H_n & \sigma < 0 \\ H_n & \sigma > 0 \end{cases}$$

Retomando el sistema (A.1), y utilizando el controlador descrito en (A.4), es posible reescribir todo el conjunto como:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= H_n \operatorname{sgn}(\sigma) + f(x_1, x_2, t) \end{aligned} \tag{A.5}$$

y sustituyendo (A.2) en (A.3) y esto a su vez en (A.5) se obtiene:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= H_n \operatorname{sgn}[k_0(x_{1ref} - x_1) + (\dot{x}_{1ref} - \dot{x}_1)] + f(x_1, x_2, t) \end{aligned} \tag{A.6}$$

El sistema obtenido en (A.6) corresponde a la forma básica dada por (A.1) la cual sigue una *referencia deseada* utilizando un controlador por modos deslizantes.

Tomando en cuenta la información anterior, se realiza una simulación del sistema utilizando el software *MATLAB R2015a/Simulink*. Los datos utilizados en la simulación se presentan a continuación:

Dato	Valor
Ganancia H_n	1.5
Ganancia k_0	0.9
Referencia deseada	4 m

Tabla A.1: Valores de la simulación del sistema de segundo orden.

El diagrama de bloques del sistema es realizado en *Simulink* y se presenta en la Figura A.2.

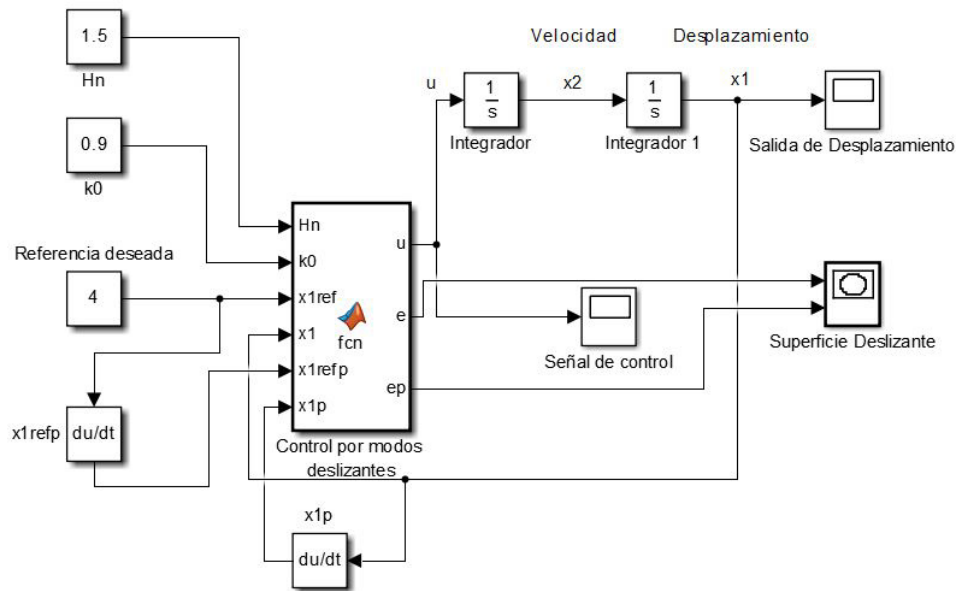


Figura A.2: Diagrama de bloques en el entorno de *Simulink*.

En la Figura A.3 se puede apreciar el buen desempeño del controlador para poder mantener el sistema en la referencia dada. Sin embargo, en la Figura A.4, que corresponde a la señal de control, se observa el cambio de la magnitud de la señal cambiando a alta frecuencia, la cual para sistemas reales, como mecánicos, no es conveniente ni factible de generar.

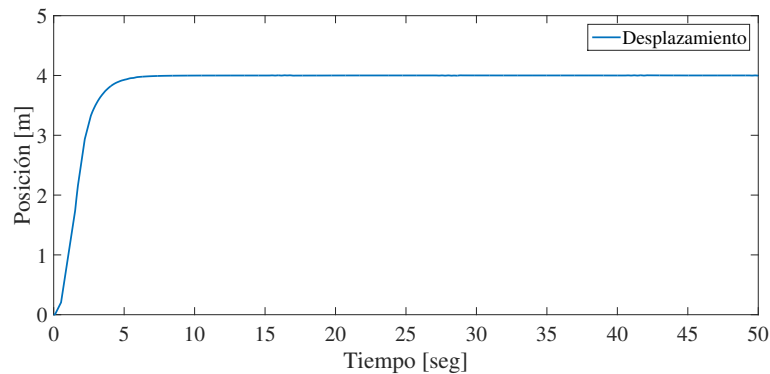


Figura A.3: Salida de desplazamiento.

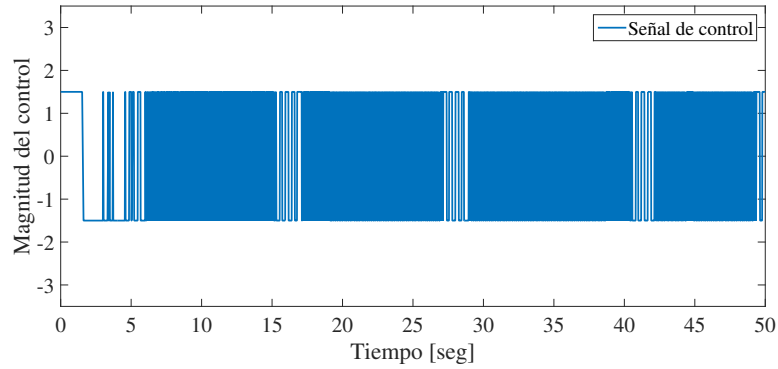


Figura A.4: Señal de control conmutando entre valores positivos y negativos.

Específicamente, si se realiza un diagrama de fase para visualizar el comportamiento de la superficie deslizante, es posible notar los efectos de tal señal de control cambiante a alta frecuencia, ver Figura A.5

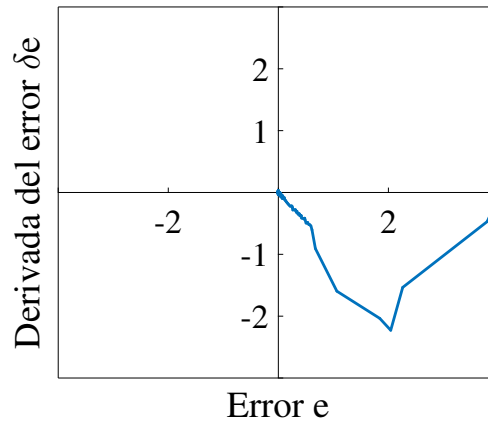


Figura A.5: Superficie Deslizante.

Además es posible observar el efecto del cambio de alta frecuencia en el área donde ya se alcanzó la línea de la superficie deslizante, tal efecto es conocido como *chattering*³(*castaño*), ver ampliación de la zona donde aparece este efecto en la Figura A.6.

³Término del idioma inglés referente a la acción de vibrar o cambiar de estado rápidamente, el termino más cercano en el idioma español es *castaño*.

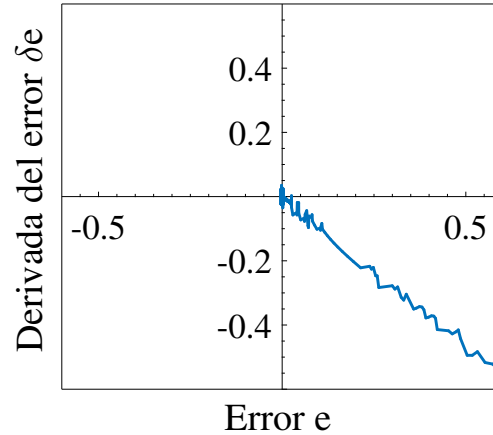


Figura A.6: Acercamiento de la zona donde se parecencia el efecto del *castaño*.

Debido a lo anterior, no es posible implementar este tipo de controlador sistemas reales sin primero eliminar el efecto de la conmutación finita de la señal de control. Es por ello que se necesita encontrar una ley de control, que a diferencia de la descrita en (A.4), sea continua en el tiempo.

A.2 Eliminación del fenómeno de *castaño*

Como se describió con anterioridad, el uso de una señal de control como la descrita en (A.4), provee un rápido y robusto desempeño en el control de sistemas; sin embargo, como se puede ver en la Figura A.4, la señal de control producida tiene una alta frecuencia de cambio, lo que hace poco factible su implementación en sistemas realistas. Este tipo de respuesta ocasiona el efecto de *castaño* lo cual es visto en la Figura A.6.

Para resolver el problema anterior, y hacer que la ley de control sea una función continua en el tiempo que evite el efecto de *castaño*, se presentan soluciones vistas en la literatura como:

- **Saturación:** Es propuesto reemplazar la función *signo* por una *saturación* para de esta manera eliminar la discontinuidad. Dado lo anterior la ley de control sería modificada como:

$$u = -H_n \text{sat}(\sigma)$$

donde la función de saturación es descrita tal que:

$$\text{sat} = \frac{\sigma}{|\sigma| + \epsilon}$$

y además es necesario elegir una constante ϵ mayor que 0 pero lo suficientemente cercano a este para no alterar la forma de la función.

- **Tangente hiperbólica:** De la misma manera, se reemplaza la función *signo* por una función de *tangente hiperbólica* y así poder eliminar la discontinuidad. La ley de control resultante es descrita como:

$$u = H_n \tanh\left(\frac{\sigma}{\epsilon}\right)$$

Las dos técnicas anteriores funcionan para cuando el sistema es estable y no se encuentra sometido a perturbaciones. Por lo tanto la convergencia de la superficie deslizante es a una *vecindad* del origen. Por esta razón es necesario encontrar una ley de control continua que de igual manera garantice la convergencia a 0 de la superficie deslizante sin perder las propiedades esenciales del controlador. Una técnica para lograr dicho cometido se presenta a continuación.

A.3 Técnica de control *Super Twisting*

Como se trató en secciones anteriores, tener una señal de control discontinua con alta frecuencia de cambio no es una opción factible en aplicaciones reales, por ello se busca tener tal señal de control como una función continua. Dado lo anterior, la técnica de control *Super-Twisting*⁴⁵ es comúnmente utilizada para poder evitar el

⁴Alusivo a la acción de girar o retorcerse.

⁵Recibe el nombre debido al comportamiento de la superficie deslizante en el plano de fase, la cual tiende a «girar» en torno al origen hasta converger a cero.

ya mencionado fenómeno de *castaño*, mientras que se mantienen las propiedades de robustez, rapidez y convergencia a cero de la superficie deslizante. La ley de control en cuestión es propuesta en [28] y es descrita como :

$$\begin{aligned} u &= c_n \sqrt{|\sigma|} \operatorname{sgn}(\sigma) + w \\ \dot{w} &= b[\operatorname{sgn}(\sigma)] \end{aligned} \tag{A.7}$$

donde w es un término de *integración* y c_n una ganancia dada por:

$$\begin{aligned} b &= 1.1H_n \\ c_n &= 1.5\sqrt{H_n} \end{aligned}$$

Para poder visualizar de manera gráfica el comportamiento de esta técnica de control, se realiza la simulación del mismo utilizando el sistema ya descrito en (A.1), donde las ganancias utilizadas se presentan en la siguiente tabla:

Dato	Valor
Ganancia H_n	0.65
Ganancia k_0	0.6
Referencia deseada	4 m

Tabla A.2: Valores de la simulación del sistema de segundo orden utilizando *Super Twisting*.

Para el caso de esta simulación, la respuesta de desplazamiento tiene un comportamiento similar que cuando se analizó el control por modos deslizantes clásico. Esto se puede ver en la Figura A.7

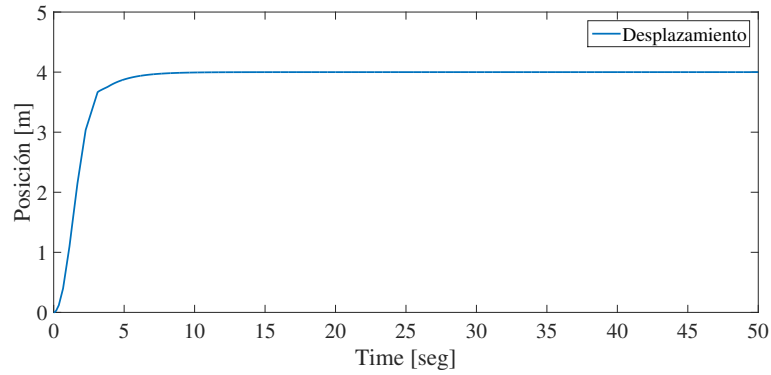


Figura A.7: Salidad de desplazamiento con el controlador *Super Twisting*.

A diferencia del controlador por modos deslizantes clásico, la principal ventaja del algoritmo de *Super Twisting* es la ley de control no es discontinua, lo que prácticamente elimina el efecto de castaño, esto se puede apreciar en la Figura A.8.

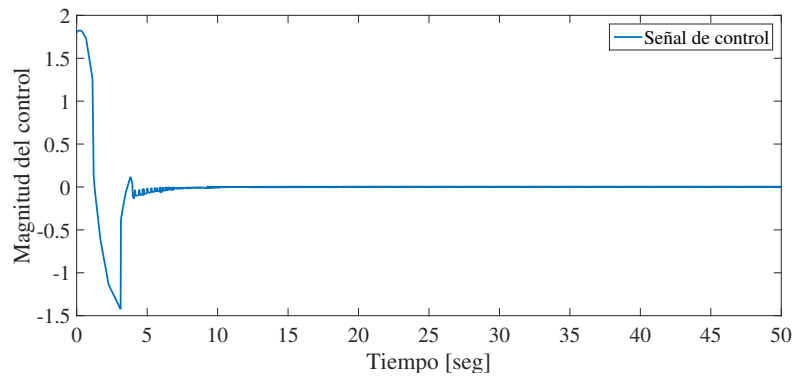


Figura A.8: Señal de control producida por la técnica de *Super Twisting*.

Finalmente, en la Figura A.9 es posible observar la evolución de la superficie deslizante, y sobretodo notar que no ocurre el fenómeno de *castaño*, por lo que resulta factible la implementación de este tipo de controlador en sistemas físicos.

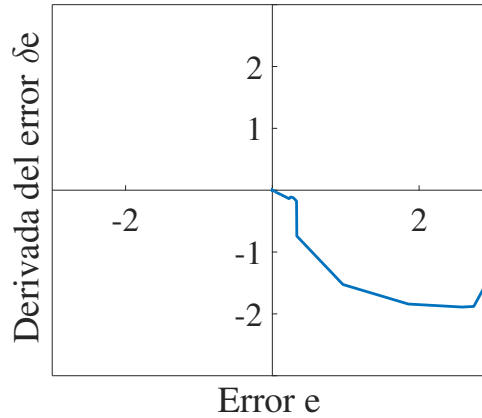


Figura A.9: Superficie deslizante producida por la ley de control *Super Twisting*.

El comportamiento principal del controlador en cuestión, tiende a hacer girar la superficie deslizante en el plano de fase hasta alcanzar la convergencia a cero, esto de igual manera puede notarse en la Figura A.9

Apéndice B

Hipervínculos de los videos de la experimentación

Todos los experimentos realizados en este trabajo fueron capturados en video. A continuación se organizan los hipervínculos para cada una de las pruebas realizadas.

B.1 Pruebas del control *Super Twisting*

- Trayectoria circular.
<https://youtu.be/cAZUM3N0kcg>
- Trayectoria en espiral.
<https://youtu.be/Ffr3Jc1IERY>
- Respuesta a perturbaciones físicas.
<https://youtu.be/nIIcxWLOAXc>

B.2 *Consensus* de segundo orden

- Trayectoria circular con dos agentes (con perturbación).

<https://youtu.be/7TM9PK1xEZc>

- Trayectoria para cubrir un área cuadrada.

<https://youtu.be/7CHxNxESAFk>

B.3 Propuesta de *Consensus no-lineal* en ley de control *Super Twisting*

- Trayectoria circular con dos agentes (con perturbación).

<https://youtu.be/cQ6UMtUKdDM>

- Trayectoria para cubrir un área cuadrada.

https://youtu.be/01M_fdWLpEo

- Trayectoria en forma de *lemniscata* con tres agentes en persecución.

<https://youtu.be/Ijqo0qU25vY>

Bibliografía

- [1] Dongbing Gu. A Differential Game Approach to Formation Control. *Transactions on control systems technology*, 16(1):85–93, 2008.
- [2] A. Bircher, M. Kamel, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart. Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots. *Autonomous Robots*, 2016.
- [3] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes. Uniform Coverage Structural Inspection Path-Planning for Micro Aerial Vehicles. *IEEE Multiconference on Systems and Control, Sydney NSW*, 2015.
- [4] A. Bircher, K. Alexis, M. Buri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart. Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics. *IEEE International Conference on Robotics and Automation ICRA, Seattle, WA*, 2015.
- [5] Y. Cao and W. Ren. Distributes Coordinated Tracking via Variable Structure Approach- Part I: Consensus Tracking. *American Control Conference , Baltimore, MD, USA*, 2010.
- [6] Z.G. Hou, L. Cheng, and M. Tan. Decentralized Robust Adaptive Control for Multiagent Systems Consensus Problem Using Neural Networks. *IEEE Transactions on Systems, Man and Cybernetics, Part: Cybernetics*, 39(3), 2009.
- [7] Z. Li and Z. Duan. Cooperative Control of Multi-agent Systems: A Consensus Region Approach. *CRC Press*, 2014.

-
- [8] W. Ren. Consensus strategies for cooperative control of vehicle formations. *IET Control Theory Appl*, 1(2), 2007.
 - [9] W. Ren and R. W. Beard. *Distributed consensus in multi-vehicle formations*. Springer-Verlag, London, 2008.
 - [10] Zhongjie Lin and Hugh Hong Tao Liu. Consensus based on learning game theory with a UAV rendezvous application. *Chinese Journal of Aeronautics*, 28(1):191–199, 2015.
 - [11] Manuel Jiménez. Zero-Sum Nonlinear Polynomial Game for Planar Robots Coordination. 2015.
 - [12] Jiménez-L, Manuel Izárraga, Ricardo Chapa, Celeste Rodriguez, Hever Arellano, and Pedro Castillo. Robust Control for Multi-Model Planar Robots Coordination. pages 899–904, 2015.
 - [13] Simon Parsons. Game Theory and Decision Theory in Multi-Agent Systems. pages 1–14, 2000.
 - [14] Dean Richert and Jorge Cort. Optimal leader allocation in UAV formation pairs under costly switching. pages 0–5, 2012.
 - [15] F. Munoz, E.S. Espinoza-Quesada, Hung La, M., S. Salazar, S. Commuri, and L. R. Garcia Carrillo. No Adaptive consensus algorithms for real.time operation of multi-agent systems affected by switching networks events. *International Journal od Robust and Nonlinear Control*, 2016.
 - [16] Robert F. Stengel. *Flight Dynamics*. Princeton University Press, USA, 2004.
 - [17] A. Abdessameud and A. Tayebi. Motion Coordination for VTOL Unmanned Aerial Vehicles: Attitude Synchronisation and Formation Control. *Springer. Verlag*, 2013.
 - [18] J. G. Leishman. *Principles of Helicopter Aerodynamics*. Cambridge University Press, USA, 2006.

-
- [19] Rogelio Lozano. *Unmanned Aerial Vehicles Embedded Control*. John Wiley-ISTE Ltd, USA, 2010.
 - [20] A. Sanchez, V. Parra-Vega, C. Izaguirre, and O. Garcia. Position-Yaw Tracking of Quadrotors. *Journal of Dynamic Systems, Measurement and Control*, 137, 2015.
 - [21] Frank L Lewis, Hongwei Zhang, Kristian Hengster-Movric, and Abhijit Das. *Cooperative Control of Multi-Agent Systems*. Springer, New York, 2014.
 - [22] Craig W Reynolds. Flocks , Herds , and Schools : A Distributed Behavioral Model. 21(4):25–34, 1987.
 - [23] Stephane Piskorski, Nicolas Brulez, Pierre Eline, and Frederic D’Haeyer. AR. Drone Developer Guie SDK 2.0, 2012.
 - [24] E. G. Rojo-Rodriguez, E. J. Ollervides, J. G. Rodriguez, E. S. Espinoza, P. Zambrano-Robledo, and O. Garcia. Implementation of a Super Twisting controller for distributed formation flight of multi-agent systems based on consensus algorithms. *International Conference of Unmanned Aircraft Systems ICUAS*, pages 1101–1107, 2017.
 - [25] Y. Zhou, X. Dong, G. Lu, and Y. Zhong. Time-varying formation control for unmanned aerial vehicles with switching interaction topologies. *International Conference of Unmanned Aircraft Systems ICUAS*, 2014.
 - [26] X. Dong, Z. Shi, G. Lu, and Y. Zhong. Time-varying output formation control for high-order linear time-invariant swarm systems. *Information Sciences*, 2015.
 - [27] J. Dennis Lawrence. *A catalog of special plane curves*. Dover Publications, Inc., New York, 1 edition, 1972.
 - [28] Yuri Shtessel, Christopher Edwards, Leonid Fridman, and Arie Levant. *Sliding Mode Control and Observation*.

-
- [29] Dipartimento Di Ingegneria Elettrica Ed Elettronica. A quick introduction to sliding mode control and its applications.
 - [30] Vadim I. Utkin. *Sliding Modes in Control and Optimization*. Springer-Verlag, first edition, 1992.
 - [31] Arie Levant. Sliding Order and Sliding Accuracy in Sliding Mode Control. *International Journal of Control*, (May), 2014.

Resumen autobiográfico

Erik Gilberto Rojo Rodríguez

Candidato para obtener el grado de
Maestría en Ingeniería Aeronáutica
con orientación en Dinámica de Vuelo

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO PARA EL VUELO
EN FORMACIÓN DE VEHÍCULOS AÉREOS NO TRIPULADOS

El Ing. Erik Gilberto Rojo Rodríguez nació el 4 de Junio de 1993 en la ciudad de Monterrey, Nuevo León, México. Hijo de Gilberto Rojo Guerrero y María de la Luz Rodríguez Díaz.

Concluyó sus estudios de bachillerato bilingüe progresivo en la Preparatoria No. 7 de la UANL en el año de 2010.

Inició sus estudios de licenciatura en *Ingeniería en Mecatrónica*, en la Facultad de Ingeniería Mecánica y Eléctrica, de la misma universidad, en el año 2010.

Laboró para la Empresa *Conmet* como asistente de mantenimiento, donde se desarrolló profesionalmente en los ámbitos de la electrónica y el control industrial.

Durante el periodo 2014-2015 fue miembro del Laboratorio de Manufactura Avanzada del Centro de Investigación e Innovación en Ingeniería Aeronáutica de la FIME-UANL, donde desarrolló distintos proyectos y recibió certificaciones de operación de equipo de medición de alta precisión.

Finalizó satisfactoriamente sus estudios de licenciatura en el año 2015, y recibió el Título de Ingeniero en Mecatrónica a principios de 2016.